Mathematics

HP COMPUTER CURRICULUM

Mathematical Systems

TEACHERS ADVISOR

HEWLETT PACKARD

HEWLETT PACKARD

Hewlett-Packard
Computer Curriculum Series

mathematics
TEACHER'S ADVISOR

mathematical systems

by E. A. Herber
Golden High School
Golden, Colorado

edited by Jean Danver and Christine Doerr
Hewlett-Packard

Hewlett-Packard Company
11000 Wolfe Road
Cupertino, California 95014

HP 5951-4444

April, 1973

The Hewlett-Packard Computer Curriculum Series represents the published results of a Curriculum Development project sponsored by the Data Systems Division of Hewlett-Packard Company. This project is under the directorship of Jean H. Danver.

This material is designed to be used with any Hewlett-Packard system with the BASIC programming language such as the 9830A Educational BASIC, and the 2000 and 3000 series systems.

The Hewlett-Packard Company makes no warranty, expressed or implied, and assumes no responsibility in connection with the operation of the program material attached hereto.

TABLE OF CONTENTS

INTRODUCTION

This Mathematics Set of the Hewlett-Packard Computer Curriculum Series consists of a set of a Student Lab Book and a corresponding Teacher's Advisor. It was designed to help meet the need for computer-oriented problems in mathematics providing students an opportunity to use a computer as a modeling device within a particular subject matter area.

The materials are designed for flexible use as desired by the individual instructor. The material and exercises in this unit are intended as an "enrichment" experience in the field of mathematical systems. The eleven algebraic properties which hold for the real number system are defined and then applied to three types of systems: sets, matrices, and modular systems. Basic operations for each system are presented; for example, the union and intersection of sets are discussed. The exercises deal with these basic operations and with verifying the applicability of the system properties. Obviously, the unit only scratches the surface of each topic covered, but most of the material will not be covered in your standard text. Thus, it can be used to supplement and enrich your curriculum in any fashion you choose.

The degree of difficulty of the material is dependent upon the amount of assistance which you choose to provide. With no assistance, the better student should be challenged. However, given a good deal of assistance, any second year algebra student should be able to work out the exercises with no great difficulty. The level of the material is determined by the assumption that students taking second year algebra will be quite capable as a group.

The Student Book provides text material and programming exercises for the students. There is a problem analysis, including a suggested approach and a macro flow chart, for each exercise. The mathematical concepts needed for each exercise are briefly reviewed, but you may want your students to study these in greater detail before attempting the exercises, especially if they have no background in sets and matrices. A list of possible references is included at the end of each section. These references will also provide additional problems for your better students. The Teacher's Advisor contains an example of a program to solve each exercise, micro flow charts, and a brief discussion of the important elements of the exercise. The micro flow charts should be given to the students only after they have made an attempt to solve the problem on their own.

For best results, you should study all the solutions until you are certain you have a complete grasp of the general methods. This should be done before assigning the material to the class. Generally, the exercises within each section are cumulative so that as techniques are developed they are used in subsequent exercises. Therefore, you will probably wish to proceed through the exercises in the order in which they are given.

You will undoubtedly think of different programming methods or techniques as you study the example programs. Encourage the students to do the same. There are no *approved* solutions. All solutions are acceptable if they produce the correct results. At this level, there is no need for emphasis on the efficiency of a student's program. The important question is, does it work?

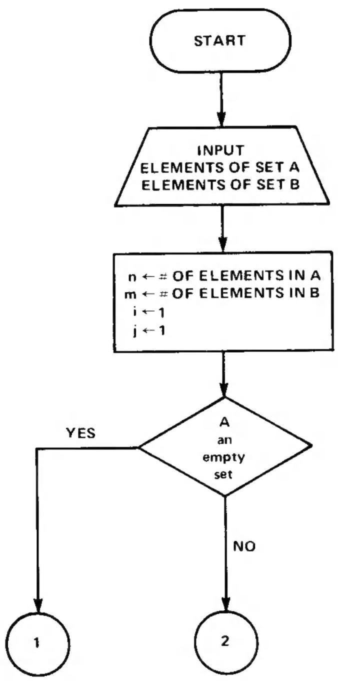
SETS

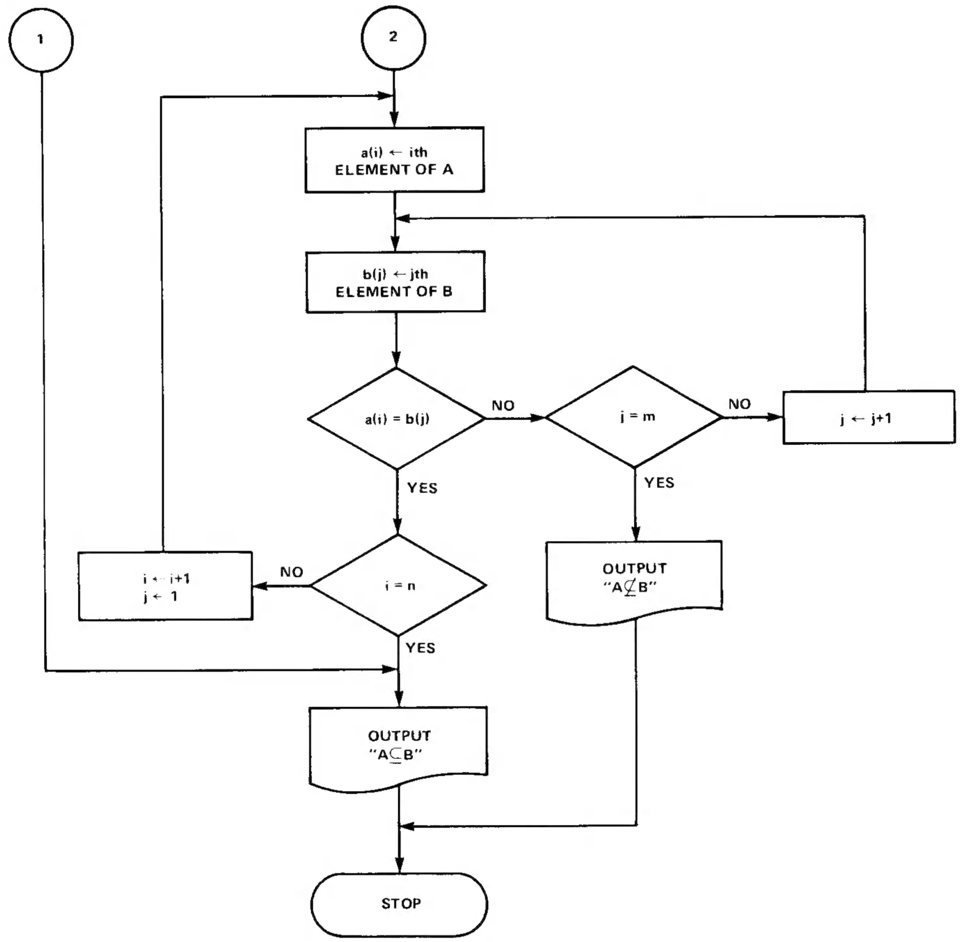
EXERCISE 1 – Determining Subsets

The procedure for this exercise is simple, but the programming is a little tricky. Three conditional loops are involved.

Micro Flow Chart

Exercise 1




1
2
a(i) ← ith
ELEMENT OF A
b(j) ← jth
ELEMENT OF B
a(i) = b(j)
NO
j = m
NO
j ← j+1
YES
YES
i ← i+1
j ← 1
NO
i = n
OUTPUT
"A⊈B"
YES
OUTPUT
"A⊆B"
STOP

Example Program

Exercise 1

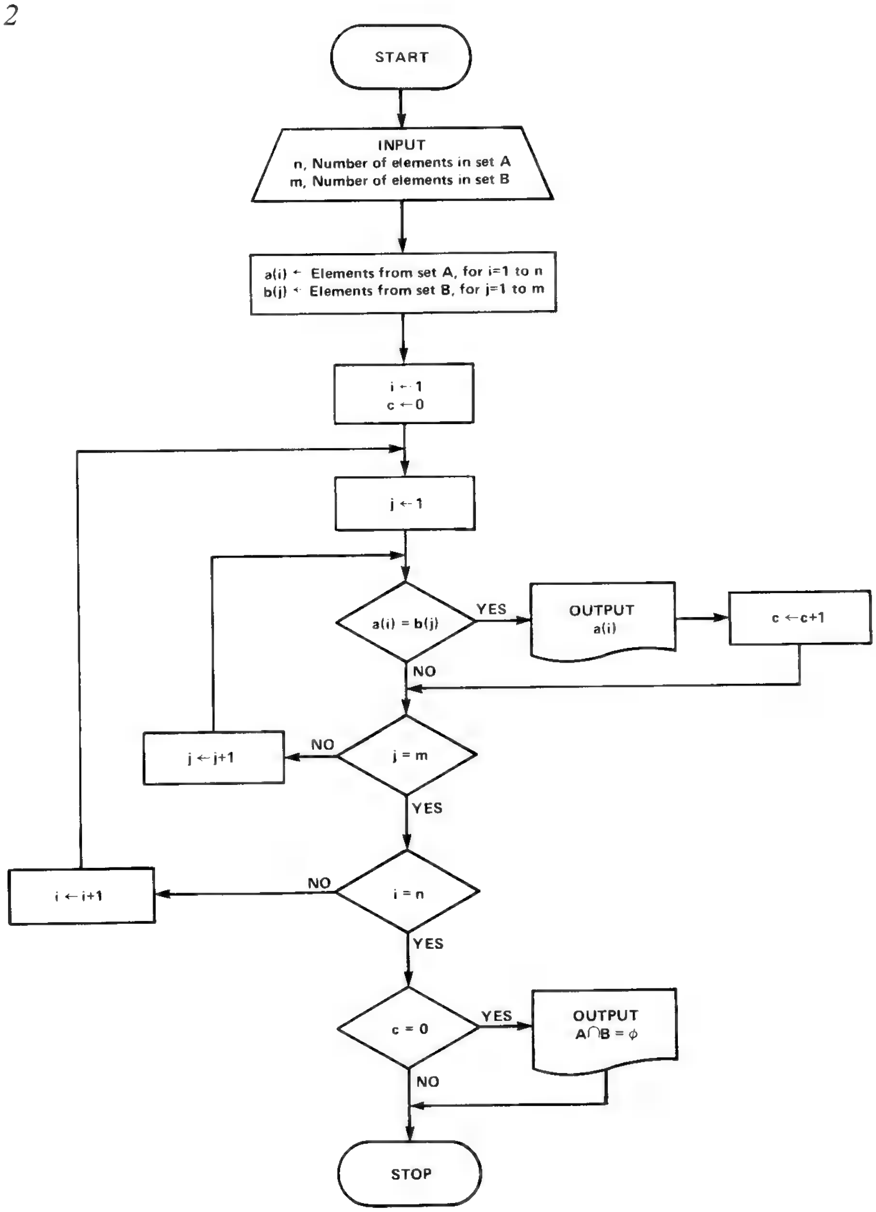
```
10  REM--THIS PROGRAM WILL DETERMINE IF A FINITE SET A OF REAL
20  REM-- NUMBERS IS A SUBSET OF A FINITE SET B.  ON THE DATA LINE
30  REM--ENTER THE NUMBER OF ELEMENTS IN EACH SET FOLLOWED BY THE
40  REM--ELEMENTS OF SET A THEN SET B.
50  READ N,M
60  IF N=0 THEN 130
70  DATA 4,6,7,16,.5,-3,.5,7,-3,13,0,16,3,3,4,.4,13,.4,4,13,5,4,-8
80  DATA 3.1415,10,1,5,-8,3.1415,10,5,4,4,17,-5,2,.01,-7,.1,3,5
90  DATA 0,2,-7,2
100  FOR I=1 TO N
110  READ A[I]
120  NEXT I
130  FOR J=1 TO M
140  READ B[J]
150  NEXT J
160  IF N=0 THEN 300
170  I=1
180  J=1
190  IF A[I]=B[J] THEN 230
200  IF J=M THEN 270
210  J=J+1
220  GOTO 190
230  IF I=N THEN 300
240  I=I+1
250  J=1
260  GOTO 190
270  PRINT "A IS NOT A SUBSET OF B"
280  PRINT
290  GOTO 50
300  PRINT "A IS A SUBSET OF B"
310  PRINT
320  GOTO 50
330  END
RUN

A IS A SUBSET OF B

A IS A SUBSET OF B

A IS NOT A SUBSET OF B

A IS NOT A SUBSET OF B

A IS A SUBSET OF B


OUT OF DATA  IN LINE 50
```

*EXERCISE 2 – **The Intersection of Sets***

This exercise models the definition of intersection. The value of the problem comes not from the significance of its application but rather from the modeling experience.

Micro Flow Chart

Exercise 2



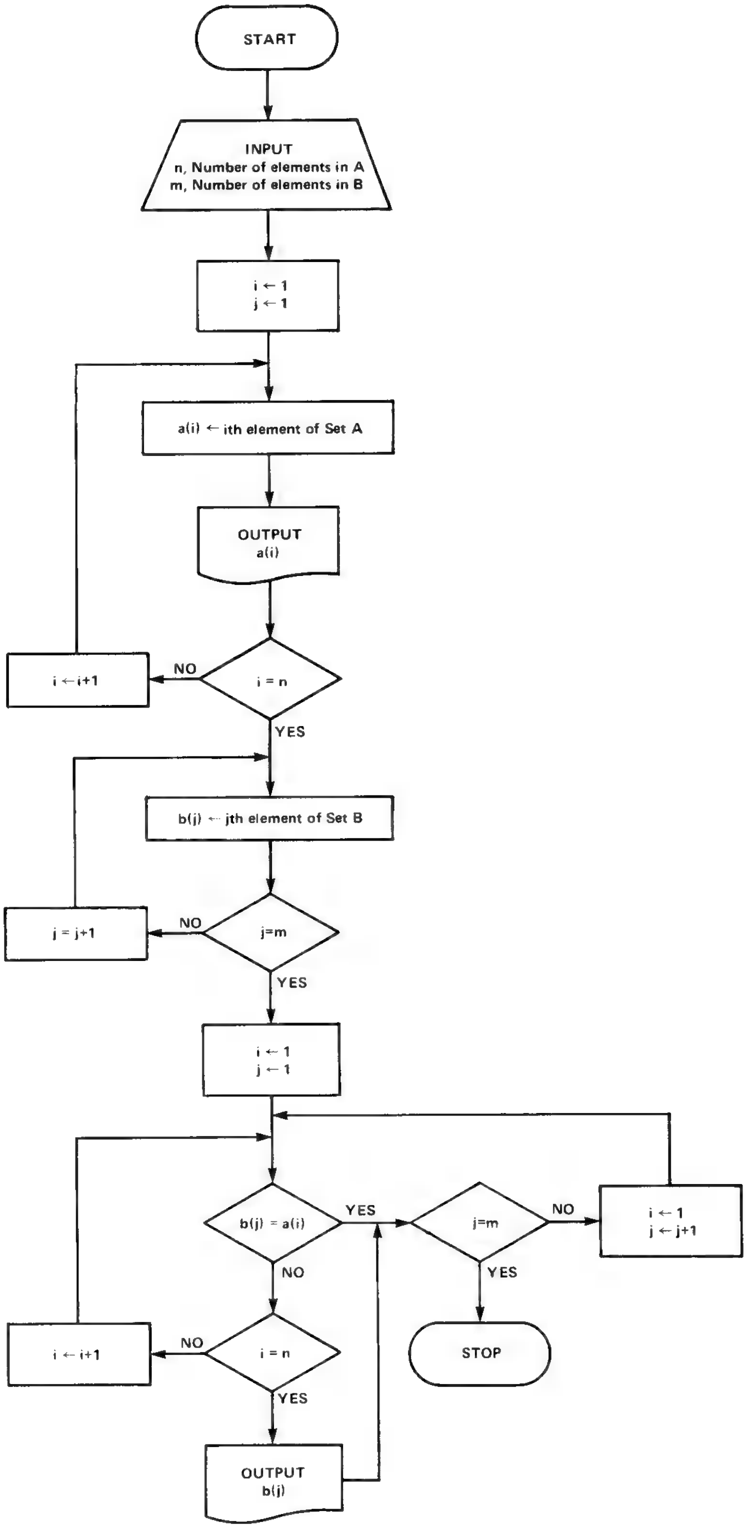
Example Program

Exercise 2

```
10  REM--THIS PROGRAM WILL DETERMINE THE INTERSECTION SET OF TWO
20  REM--NON-EMPTY SETS, A AND B, OF REAL NUMBERS.
40  LET C=0
50  DATA 4,3,-2.5,7,.27,4,-.7,3,-6,.5,3,17,3.14,-17,5,0,17,-6
60  DATA 3.14,-17,4,7,-.66,25,6,3,-2,0,2.16,3,-5,0,1,3,0,-5,1
70  READ N
75  PRINT "A INTERSECT B =";
80  FOR I=1 TO N
90  READ A[I]
100  NEXT I
110  READ M
120  FOR J=1 TO M
130  READ B[J]
140  NEXT J
150  FOR I=1 TO N
160  FOR J=1 TO M
170  IF A[I]=B[J] THEN 220
180  NEXT J
190  NEXT I
200  IF C=0 THEN 250
205  PRINT
206  PRINT
210  GOTO 40
220  LET C=C+1
230  PRINT A[I];
240  GOTO 190
250  PRINT "THE NULL SET"
251  PRINT
252  PRINT
255  GOTO 40
260  END
RUN

A INTERSECT B = 3

A INTERSECT B = 17      3.14          -17

A INTERSECT B =THE NULL SET


A INTERSECT B =-5       0       1


OUT OF DATA  IN LINE 70
```

EXERCISE 3 – The Union of Sets

Micro Flow Chart



Example Program

Exercise 3

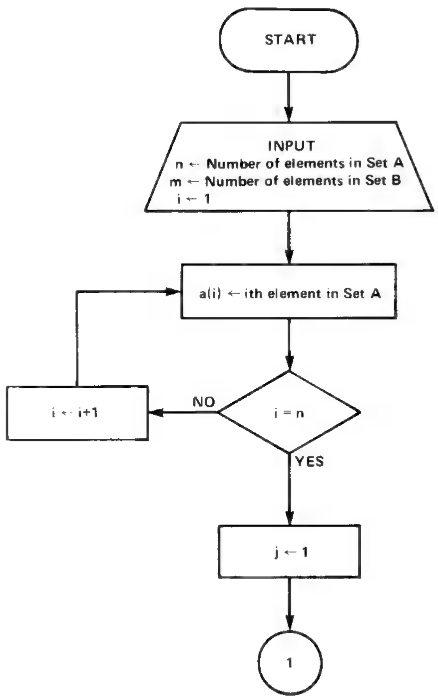
```
10  REM-- A PROGRAM TO FORM THE UNION OF TWO NON-EMPTY SETS OF
20  REM-- NUMBERS.  THE DATA LINE SHOULD CONTAIN THE NUMBER OF
40  REM-- ELEMENTS IN THE SECOND SET.  THEN THE ELEMENTS OF
50  REM-- OF THE FIRST SET AND THE ELEMENTS OF THE SECOND SET
60  REM-- IN THAT ORDER.
70  READ N,M
75  PRINT "THE UNION OF SET A WITH SET B CONTAINS ELEMENTS:";
80  FOR I=1 TO N
90  READ A[I]
95  PRINT A[I];
100  NEXT I
110  FOR J=1 TO M
120  READ B[J]
130  NEXT J
140  I=1
150  J=1
160  IF B[J]=A[I] THEN 200
170  IF I=N THEN 230
180  I=I+1
190  GOTO 160
200  IF J=M THEN 300
205  I=1
210  J=J+1
220  GOTO 160
230  PRINT B[J];
240  GOTO 200
250  DATA 5,5,6,7,-2,-1.6,75,3,-18,-2,75,11
300  END
RUN

THE UNION OF SET A WITH SET B CONTAINS ELEMENTS: 6      7      -2
-1.6          75     3     -18     11


250 DATA 5,5,7,-7,0,1,-1,7,-7,0,1,-1
RUN

THE UNION OF SET A WITH SET B CONTAINS ELEMENTS: 7     -7      0
-1
```

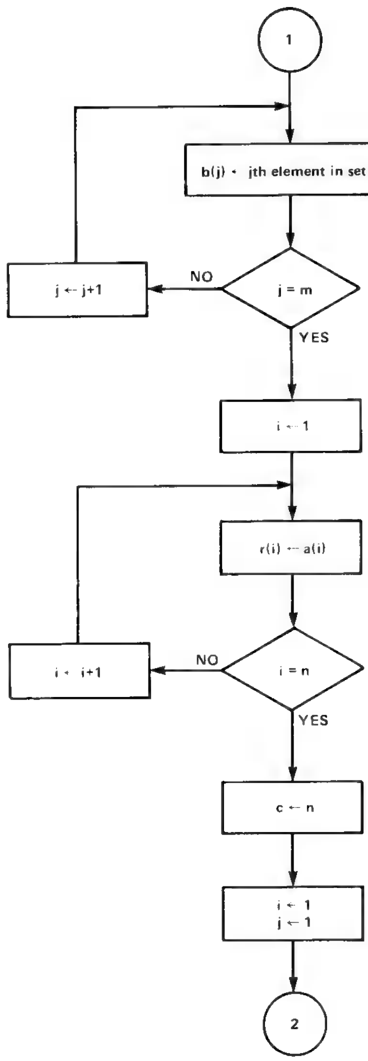
EXERCISE 4 – Modeling Commutativity of the Union Operation For Two Given Sets A and B

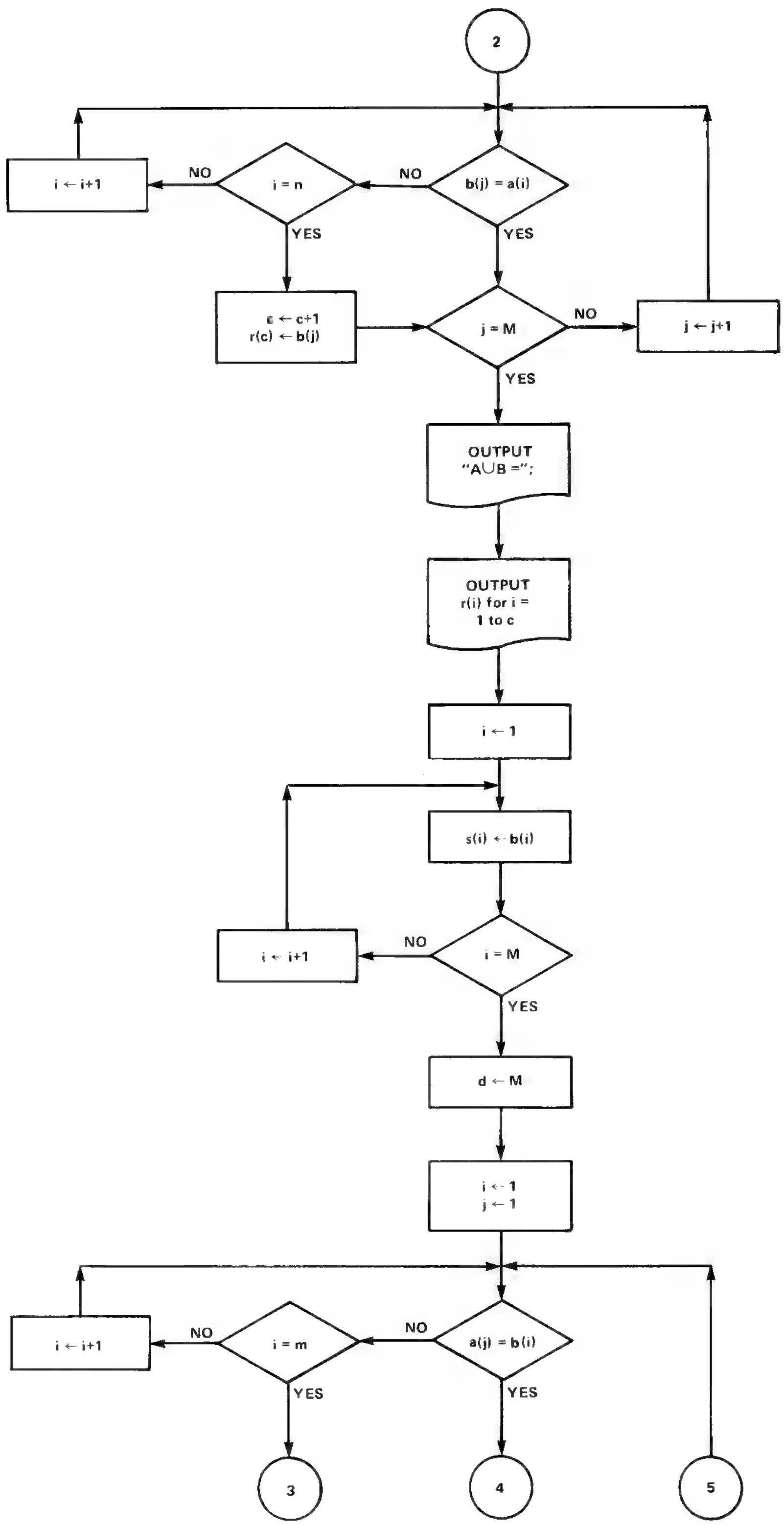
This is a good exercise in flow charting. If the flow charting is done correctly, the BASIC program easily follows.

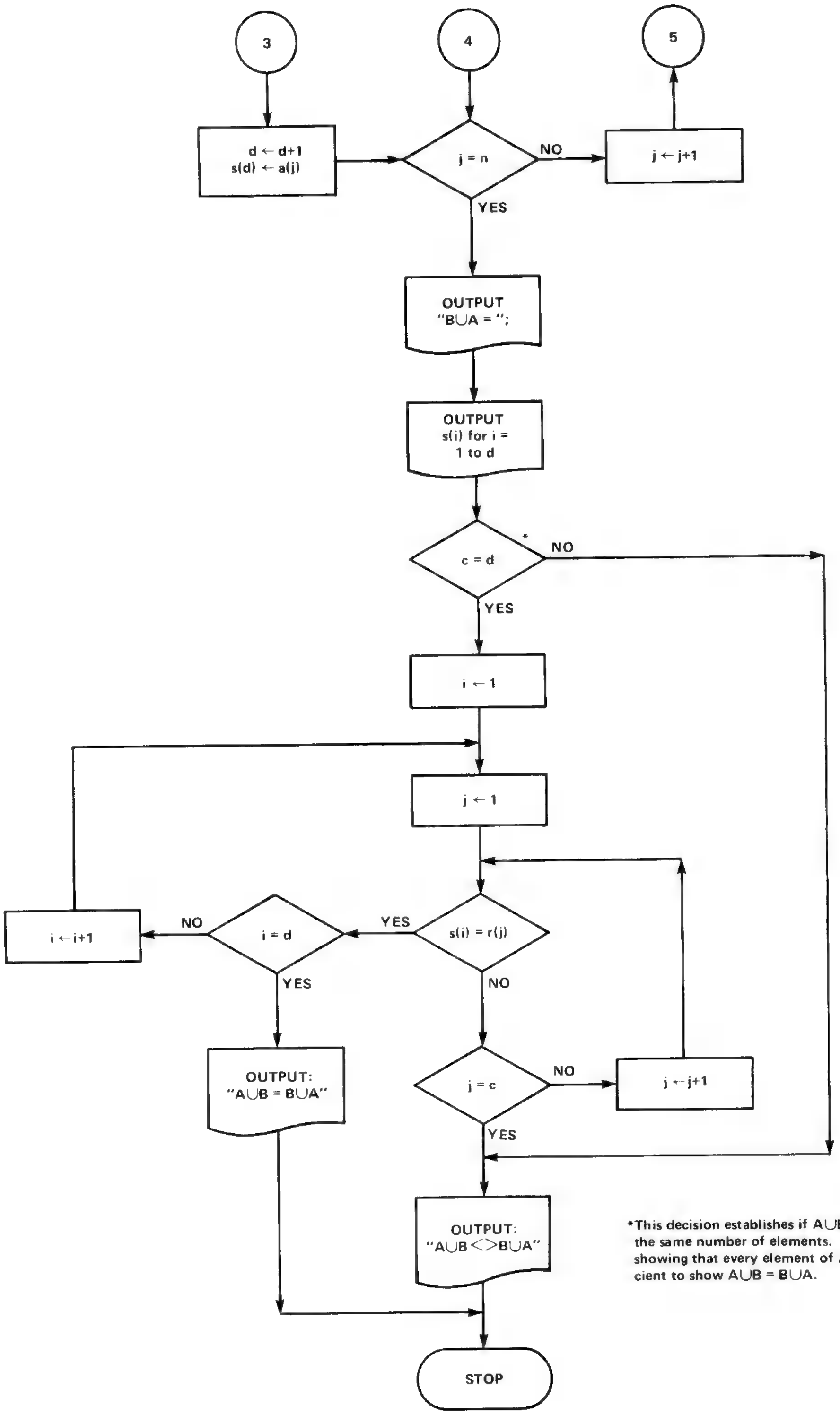
You might also point out to students that writing formal proofs of generalization is not so bad when you consider the work involved in checking this one instance. Of course, a paper and pencil check would be rather easy if the number of elements in A or B is not excessive.

Micro Flow Chart




1
b(j) ← jth element in set B
j = m
NO
YES
j ← j+1
i ← 1
r(i) ← a(i)
i = n
NO
YES
i ← i+1
c ← n
i ← 1
j ← 1
2


2
i ← i+1
NO
i = n
NO
b(j) = a(i)
YES
YES
c ← c+1
r(c) ← b(j)
j = M
NO
j ← j+1
YES
OUTPUT
"A∪B =";
OUTPUT
r(i) for i =
1 to c
i ← 1
s(i) ← b(i)
i ← i+1
NO
i = M
YES
d ← M
i ← 1
j ← 1
i ← i+1
NO
i = m
NO
a(j) = b(i)
YES
YES
3
4
5



*This decision establishes if A∪B and B∪A have the same number of elements. If they do, then showing that every element of A is in B is sufficient to show A∪B = B∪A.

Example Program

Exercise 4

```
10  REM--THIS PROGRAM DETERMINES IF THE "UNION" OPERATIONS
11  REM--ON TWO GIVEN SETS A AND B IS COMMUTATIVE.  ON THE DATA
12  REM--LINE THE OPERATOR MUST PROVIDE THE NUMBER OF ELEMENTS IN THE SE
13  REM--A, THE NUMBER OF ELEMENTS IN THE SET B, THE ELEMENTS OF SET A
14  REM--AND THE ELEMENTS OF SET B IN THAT ORDER.
20  READ N,M
30  FOR I=1 TO N
40  READ A[I]
50  NEXT I
60  FOR J=1 TO M
70  READ B[J]
80  NEXT J
90  FOR I=1 TO N
100  R[I]=A[I]
110  NEXT I
120  C=N
130  FOR J=1 TO M
140  FOR I=1 TO N
150  IF B[J]=A[I] THEN 190
160  NEXT I
170  LET C=C+1
180  LET R[C]=B[J]
190  NEXT J
200  PRINT "A U B =";
205  FOR I=1 TO C
210  PRINT R[I];
220  NEXT I
230  PRINT
231  FOR I=1 TO M
240  S[I]=B[I]
250  NEXT I
260  D=M
270  FOR J=1 TO N
280  FOR I=1 TO M
290  IF A[J]=B[I] THEN 310
300  NEXT I
303  LET D=D+1
305  LET S[D]=A[J]
310  NEXT J
315  PRINT "B U A =";
320  FOR I=1 TO D
330  PRINT S[I];
335  NEXT I
340  PRINT
350  PRINT
370  IF C=D THEN 400
380  PRINT "THEREFORE A U B <> B U A"
385  PRINT ""
390  GOTO 20
400  FOR I=1 TO D
410  FOR J=1 TO C
420  IF S[I]=R[J] THEN 450
430  NEXT J
440  GOTO 380
450  NEXT I
470  PRINT "THERFORE A U B = B U A"
473  PRINT ""
475  GOTO 20
480  DATA 5,5,6,7,-2,-1.6,75,3,-18,-2,75,11,5,5,7,-7,0,1,-1,7,-7,0,1,-1
490  END
```

```
RUN

A U B = 6      7     -2     -1.6          75     3     -18     11
B U A = 3    -18     -2      75     11     6     7     -1.6

THERFORE A U B = B U A


A U B = 7     -7      0      1     -1
B U A = 7     -7      0      1     -1

THERFORE A U B = B U A


OUT OF DATA  IN LINE 20
```

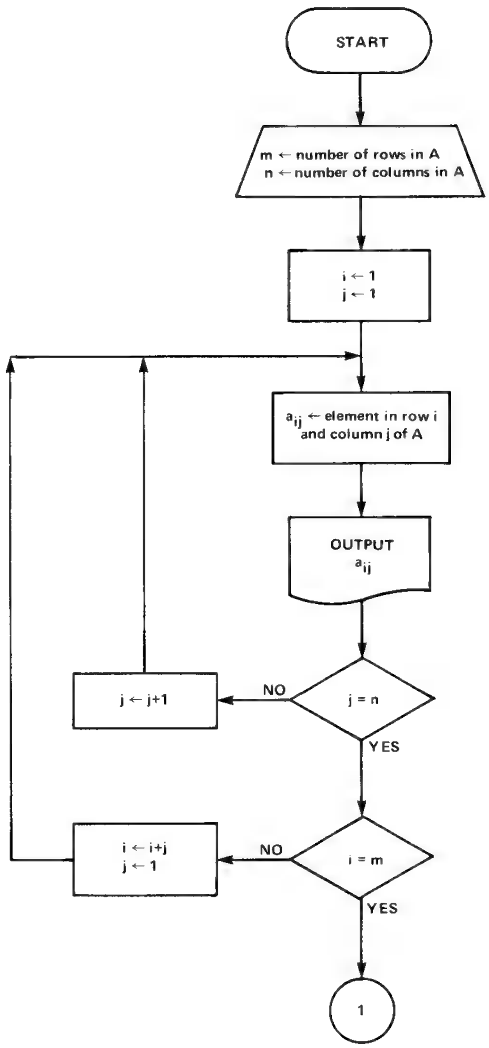
MATRICES

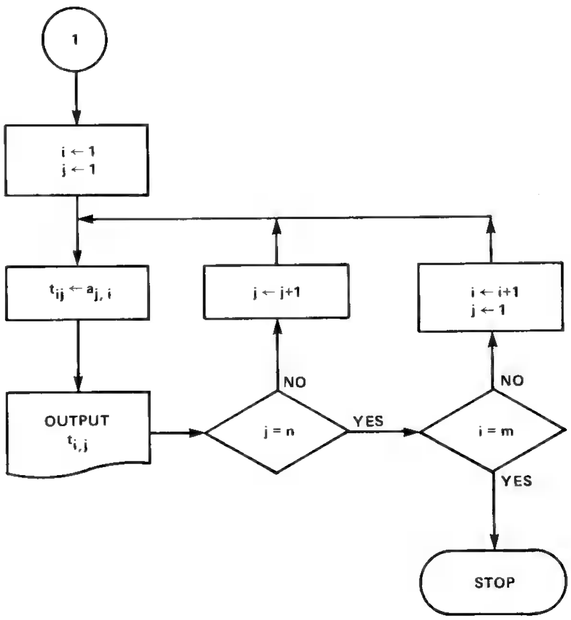
EXERCISE 5 – Modeling Matrix Definitions

This exercise tests the student's understanding of the basic definitions in matrix algebra while providing experience in manipulating elements within a matrix.

Micro Flow Chart

Exercise 5(a)





Example Program

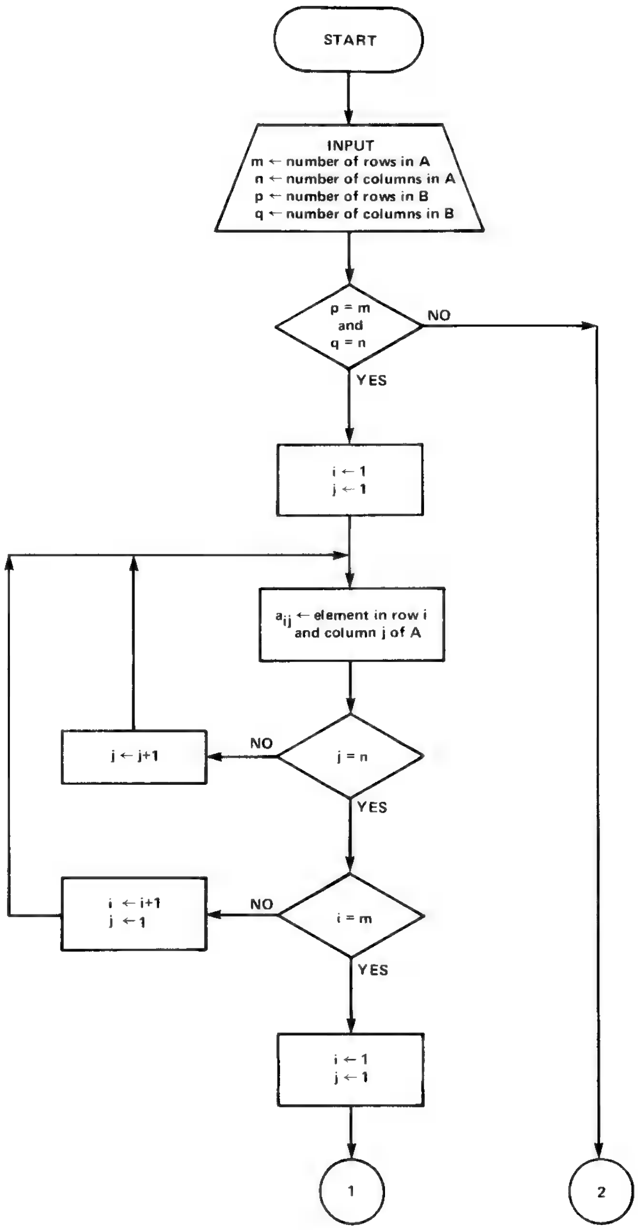
Exercise 5(a)

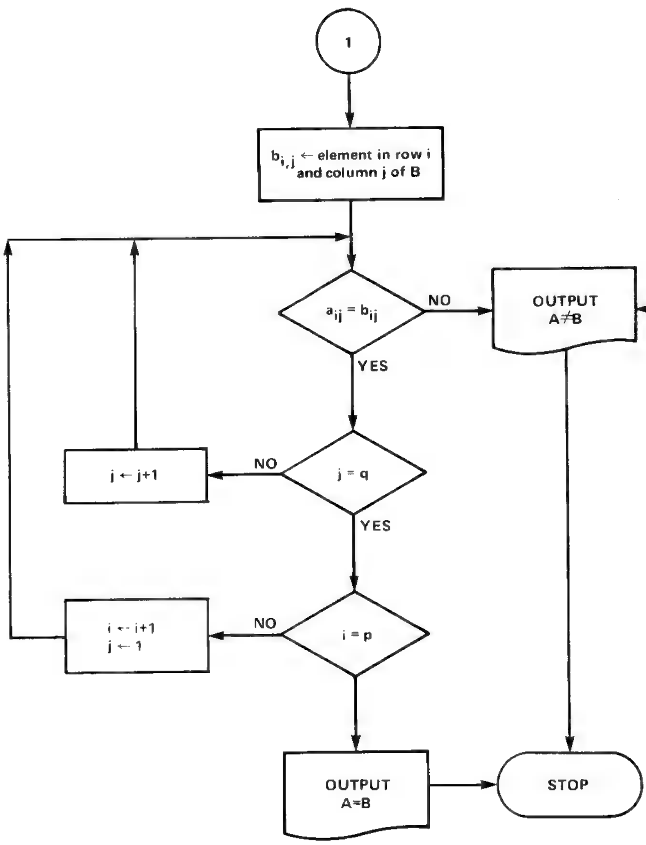
```
10 REM--A PROGRAM TO PRINT THE TRANSPOSE OF A MATRIX.
20 REM--AND THE ELEMENTS OF THE MATRIX READING FROM LEFT TO
30 REM--RIGHT ROW BY ROW
40 PRINT
50 READ N,M
60 PRINT "MATRIX A IS"
70 PRINT
80 FOR I=1 TO M
90 FOR J=1 TO N
100 READ A[I,J]
110 PRINT TAB(11*J);A[I,J];
120 NEXT J
130 PRINT
140 NEXT I
150 PRINT
160 PRINT
170 PRINT
180 PRINT "THE TRANSPOSE OF A IS"
190 PRINT
200 FOR I=1 TO M
210 FOR J=1 TO N
220 LET T[I,J]=A[J,I]
230 PRINT TAB(11*J);T[I,J];
240 NEXT J
250 PRINT
260 NEXT I
270 DATA 4,4,1,-7,0,8,3,0.5,-3,7,0,1,15,23,8,2,5,-0.8
280 END
```

```
RUN

MATRIX A IS

                1              -7              0              8
                3               0.5           -3              7
                0               1              15             23
                8               2              5             -0.8


THE TRANSPOSE OF A IS

                1               3              0              8
               -7               0.5            1              2
                0              -3              15             5
                8               7              23            -0.8

DONE
```

Micro Flow Chart

Exercise 5(b)




1
2
$b_{i,j}$ ← element in row i and column j of B
$a_{ij} = b_{ij}$
NO
YES
OUTPUT A≠B
j ← j+1
NO
j = q
YES
i ← i+1
j ← 1
NO
i = p
OUTPUT A=B
STOP

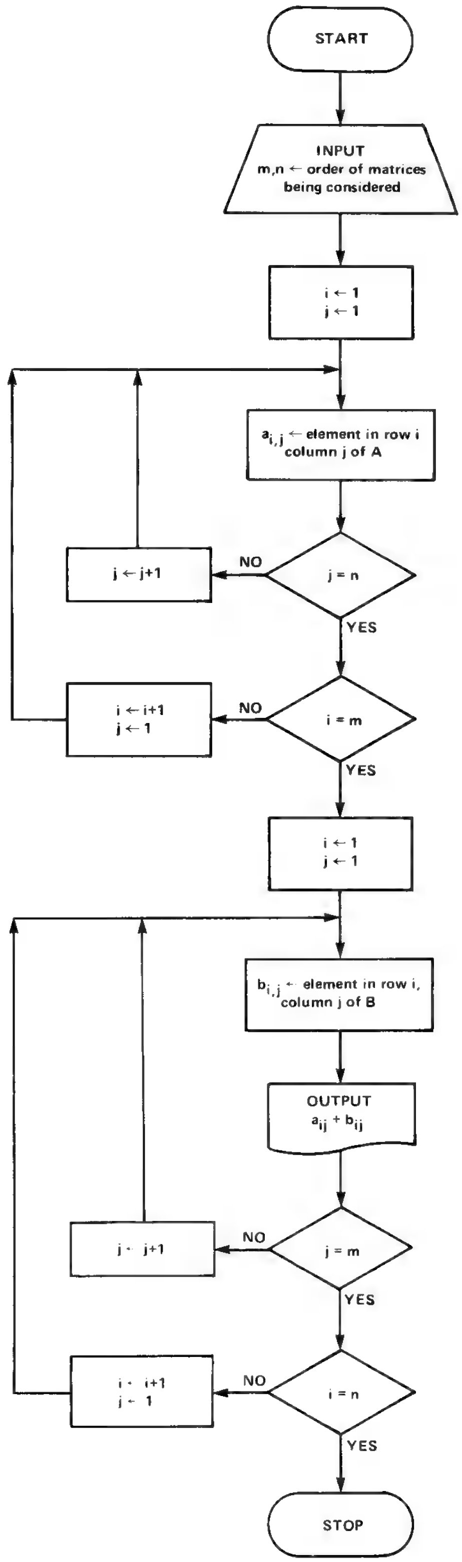
Example Program

Exercise 5(b)

```
10   REM--A PROGRAM TO DETERMINE IF TWO GIVEN MATRICES ARE EQUAL.
20   REM--ON THE DATA LINE ENTER THE ORDER M,N AND P,Q OF MATRICES
30   REM--A AND B RESPECTIVELY. THEN ENTER THE ELEMENTS OF A AND B
40   REM--ROW BY ROW.
50   READ M,N,P,Q
60   IF M <> P THEN 210
70   IF N <> Q THEN 210
80   FOR I=1 TO M
90   FOR J=1 TO N
100   READ A[I,J]
110   NEXT J
120   NEXT I
130   FOR I=1 TO P
140   FOR J=1 TO Q
150   READ B[I,J]
160   IF A[I,J] <> B[I,J] THEN 210
170   NEXT J
180   NEXT I
190   PRINT "A = B"
200   GOTO 300
210   PRINT "B <> A"
220   GOTO 240
230   DATA 2,3,3,3,3,2,-1,6,8,0,3,2,-1,6,8,0,-5,6,.6
300   END
RUN

B <> A

DONE


230 DATA 3,3,3,3,3,2,-1,6,8,0,-5,7,.6,3,2,-1,6,8,0,-5,6,.6
RUN


B <> A

DONE


230 DATA 4,4,4,4,7,-2,.5,1,6,0,-7,0,-.7,2,1,3,12,8,19,-20,7,-2
235 DATA .5,1,6,0,-7,0,-.7,2,1,3,12,8,19,-20
RUN


A = B
```

Micro Flow Chart

Exercise 5(c)



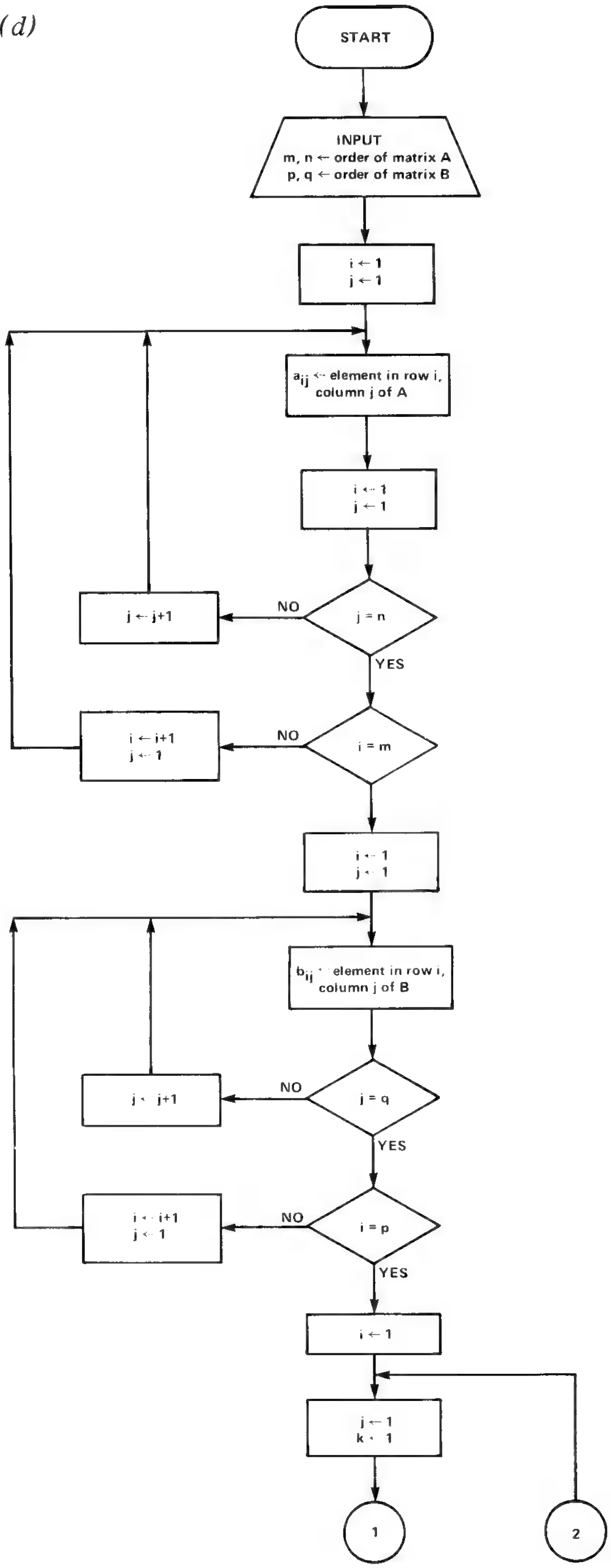
Example Program

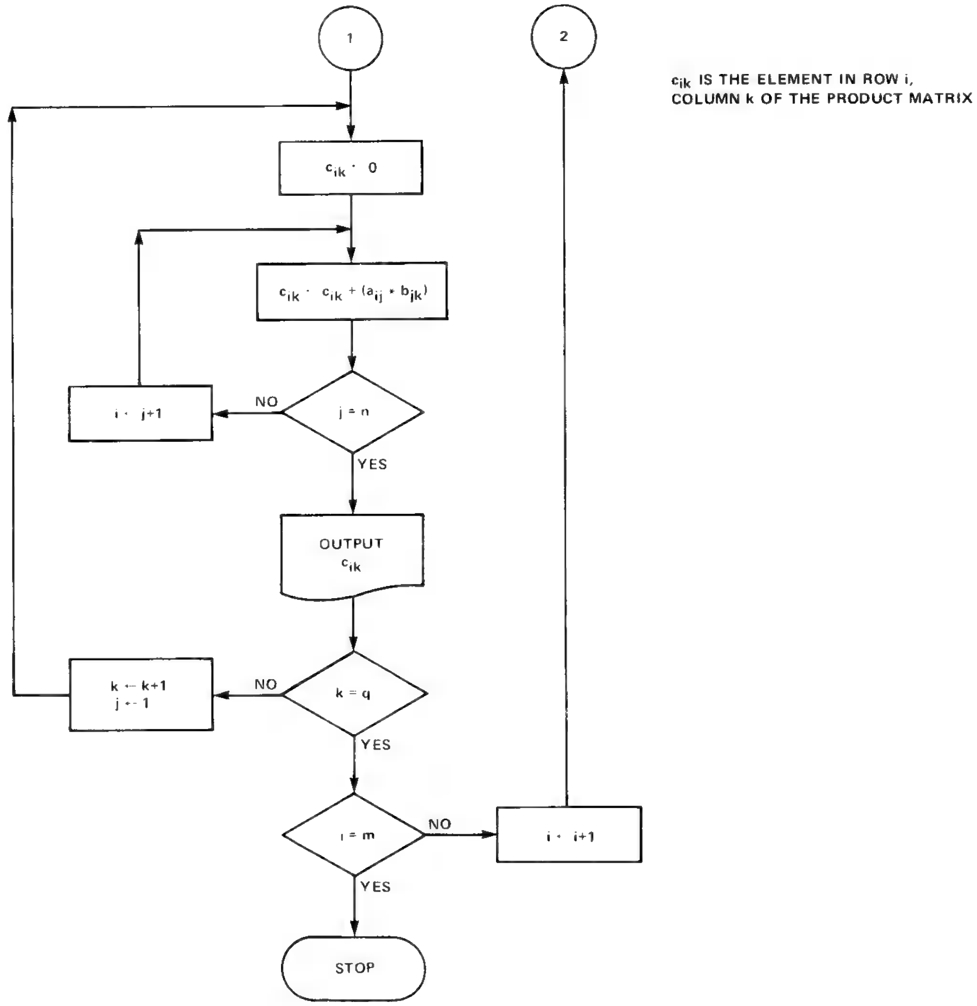
Exercise 5(c)

```
10  REM--THIS PROGRAM MODELS THE DEFINITION OF THE SUM OF TWO
20  REM--MATRICES A AND B.  ENTER ON THE DATA LINE
30  REM-- THE ORDER OF THE THE MATRICES M,N AND THE ELEMENTS OF
40  REM--OF MATRIX A AND MATRIX B READING FROM LEFT TO RIGHT
50  REM--ROW BY ROW.
60  READ M,N
70  FOR I=1 TO M
80  FOR J=1 TO N
90  READ A[I,J]
100  NEXT J
110  NEXT I
120  FOR I=1 TO M
130  FOR J=1 TO N
140  READ B[I,J]
150  PRINT A[I,J]+B[I,J];
160  NEXT J
170  PRINT
180  NEXT I
190  PRINT
200  PRINT
210  PRINT
220  GOTO 60
230  DATA 4,3,8,-2,5,-20,.5,1,0,-7,.8,6,1,0,16,27,0,-1,18,2,5
240  DATA -.16,3,0,1,7,2,4,3,-7,1,.8,2,7,0,5,17,7,0,.2,3,3,10,5
250  END
RUN

 24     25     5
-21     18.5          3
 5     -7.16          3.8
 6      2      7


 20     0      1      1
 5      10     10     10



OUT OF DATA   IN LINE 60
```

Micro Flow Chart

Exercise 5(d)




1
2
c_{ik} IS THE ELEMENT IN ROW i, COLUMN k OF THE PRODUCT MATRIX
$c_{ik} \leftarrow 0$
$c_{ik} \leftarrow c_{ik} + (a_{ij} * b_{jk})$
$j = n$
NO
$j \leftarrow j+1$
YES
OUTPUT c_{ik}
$k = q$
NO
$k \leftarrow k+1$
$j \leftarrow 1$
YES
$i = m$
NO
$i \leftarrow i+1$
YES
STOP

Example Program

Exercise 5(d)

```
10  REM--A PROGRAM TO MODEL MATRIX MULTIPLICATION.
20  REM--ON THE DATA LINE ENTER M,N AND P,Q, THE ORDER OF THE
30  REM--MATRICES A AND B RESPECTIVELY.  THEN ENTER THE ELEMENTS
40  REM--OF THE MATRICES, ROW BY ROW.
50  READ M,N,P,Q
60  PRINT
70  FOR I=1 TO M
80  FOR J=1 TO N
90  READ A[I,J]
100  NEXT J
110  NEXT I
120  FOR I=1 TO P
130  FOR J=1 TO Q
140  READ B[I,J]
150  NEXT J
160  NEXT I
170  I=1
180  J=K=1
190  C[I,K]=0
200  C[I,K]=C[I,K]+A[I,J]*B[J,K]
210  IF J=N THEN 240
220  J=J+1
230  GOTO 200
240  PRINT C[I,K];
250  IF K=Q THEN 290
260  K=K+1
270  J=1
280  GOTO 190
290  PRINT
300  IF I=M THEN 50
310  I=I+1
320  GOTO 180
330  DATA 3,3,3,3,-3,0,1,2,1,3,-2,0,1,2,1,-1,2,3,1,-2,1,0,2,3
340  DATA 3,2,3,-2,1,2,0,-5,2,-5,0,1,2,2,2,2,2,2,2,1,1,1,1,-1,-1,2
350  END
RUN


-8     -2     3
 0      8    -1
-6     -1     2

 8    -15
-6    -20

 1      0
 0      1

OUT OF DATA  IN LINE 50
```

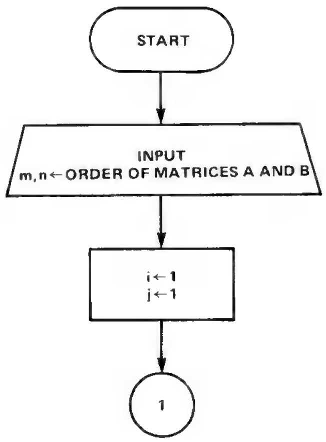
EXERCISE 6 – Properties of a System of Square Matrices

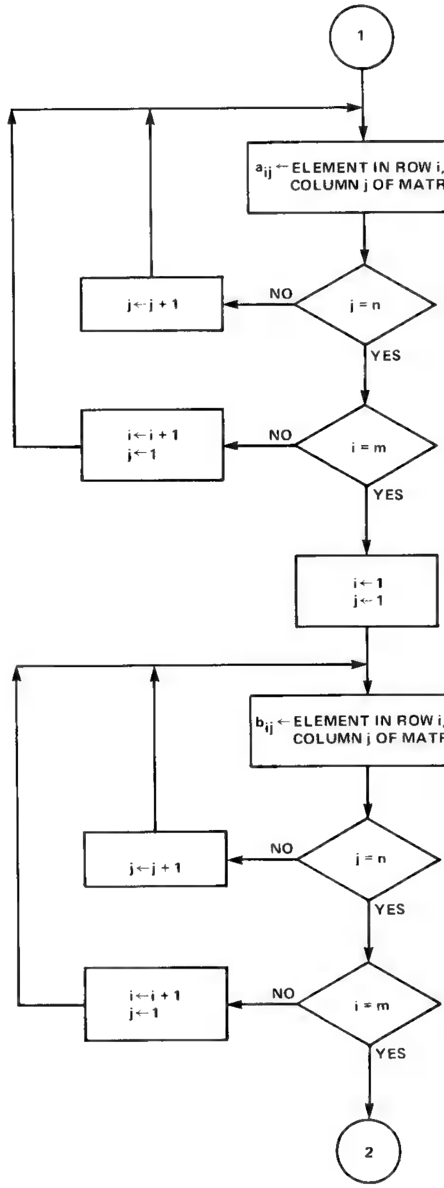
Although the computer system you are using may have matrix manipulation capabilities, is is important that your students understand the processes involved in each operation. This exercise requires that the students be familiar with the basic operations and properties of matrix algebra.

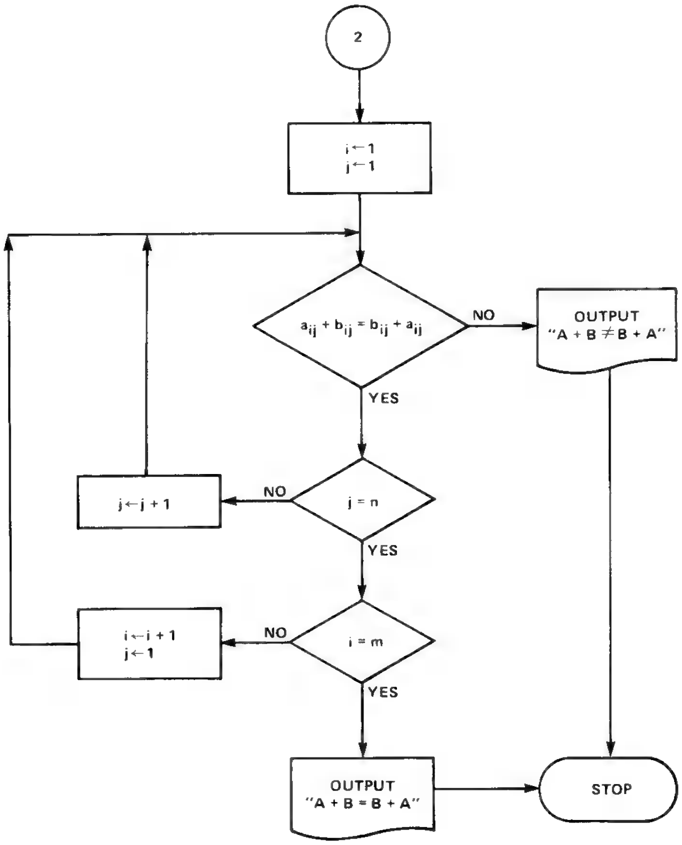
(a) In this part of the exercise, be sure to emphasize that because commutativity holds for a finite number of cases does not necessarily prove that it holds for all cases.

Micro Flow Chart

Exercise 6(a)




1
a_{ij} ← ELEMENT IN ROW i, COLUMN j OF MATRIX A
NO
j ← j + 1
j = n
YES
NO
i ← i + 1
j ← 1
i = m
YES
i ← 1
j ← 1
b_{ij} ← ELEMENT IN ROW i, COLUMN j OF MATRIX B
NO
j ← j + 1
j = n
YES
NO
i ← i + 1
j ← 1
i = m
YES
2


2
i←1
j←1
$a_{ij} + b_{ij} = b_{ij} + a_{ij}$
NO
OUTPUT
"A + B ≠ B + A"
YES
j←j + 1
NO
j = n
YES
i←i + 1
j←1
NO
i = m
YES
OUTPUT
"A + B = B + A"
STOP

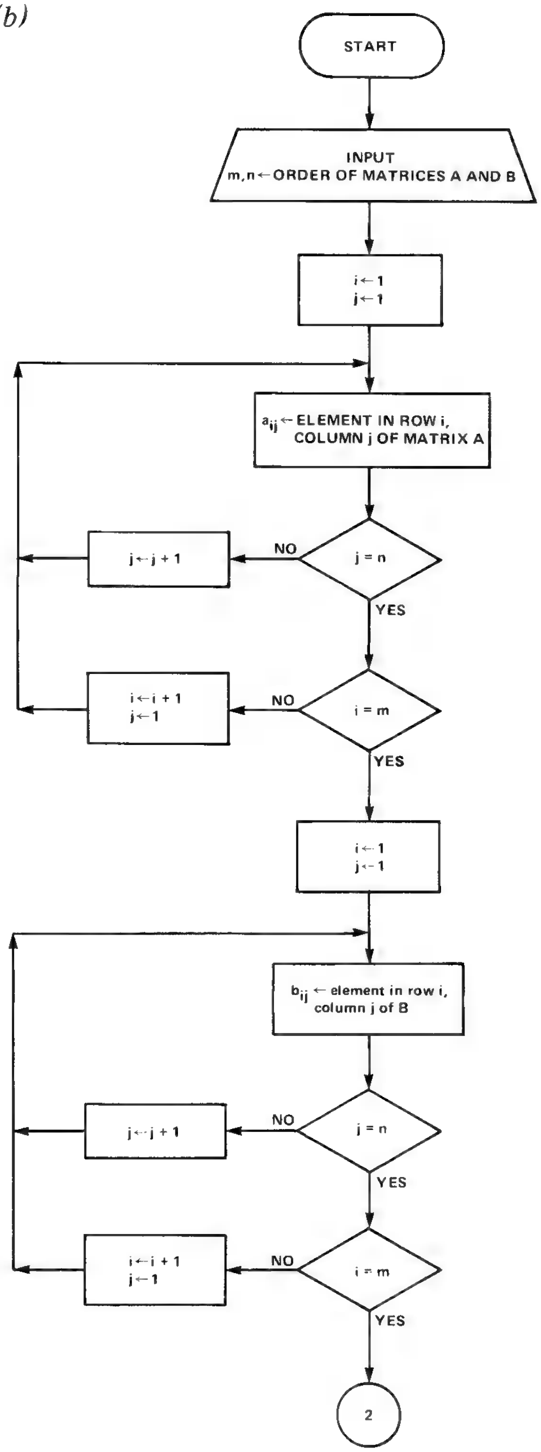
Example Program

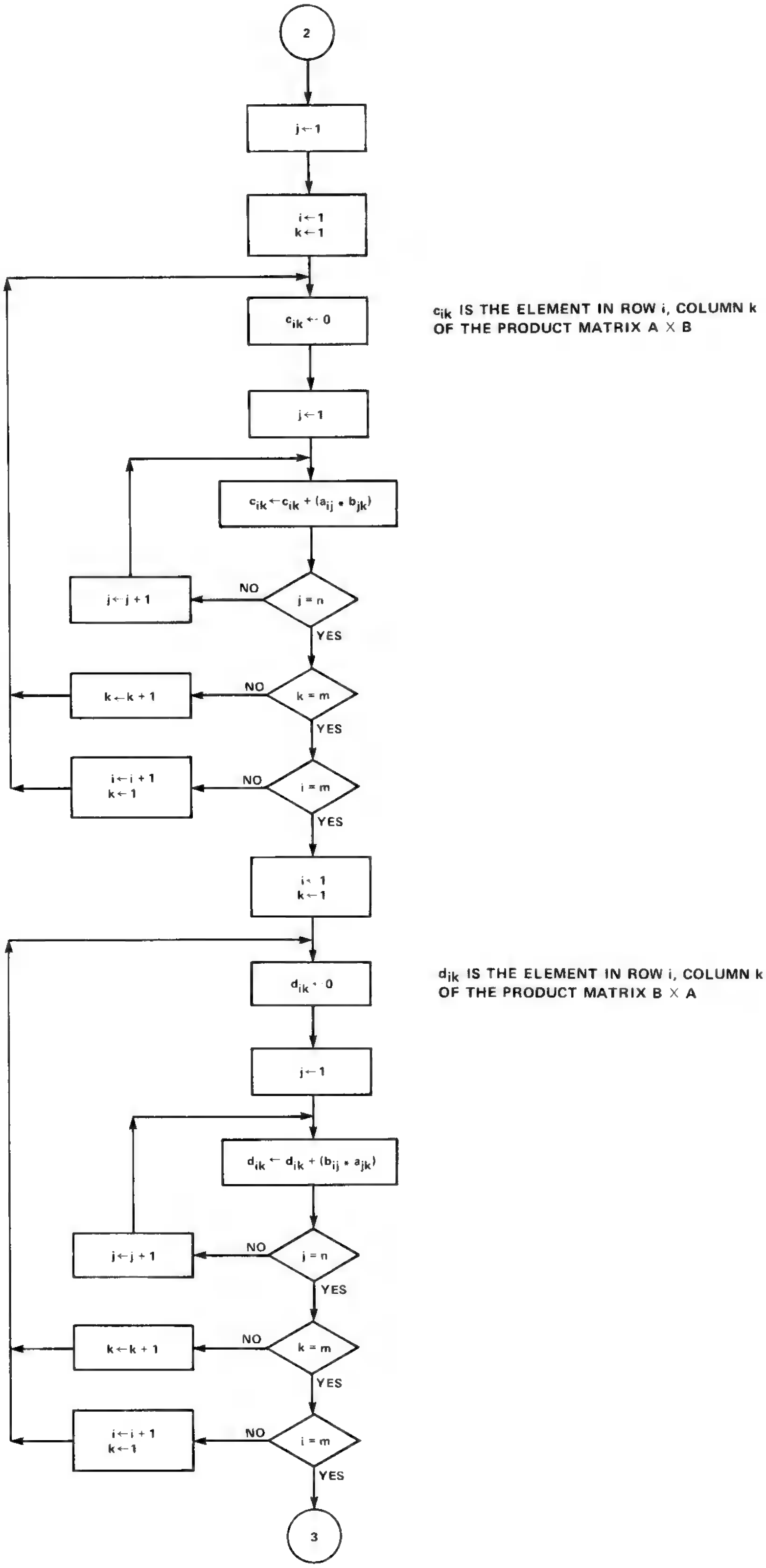
Exercise 6(a)

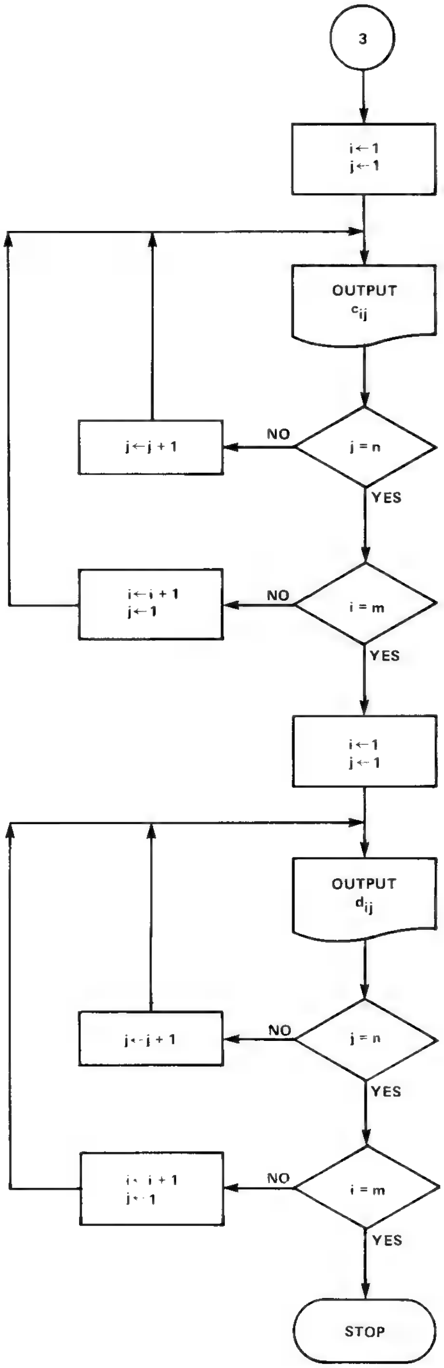
```
10  REM--THIS PROGRAM ILLUSTRATES THE COMMUNITIVE PROPERTY FOR MATRIX
20  REM--ADDITION.  AN INPUT OF THE ORDER OF THE MATRIX IS REQUIRED
30  REM--THE DATA LINE SHOULD CONTAIN THE ELEMENTS OF MATRICES A AND B
40  DATA 8,-2,5,20,.5,1,0,-7,.8,6,1,0
50  DATA 16,27,0,-1,18,2,5,-.16,3,0,1,7
60  PRINT "WHAT IS THE ORDER OF THE MATRICES INVOLVED";
70  INPUT M,N
80  FOR I=1 TO M
90  FOR J=1 TO N
100  READ A[I,J]
110  NEXT J
120  NEXT I
130  FOR I=1 TO M
140  FOR J=1 TO N
150  READ B[I,J]
160  NEXT J
170  NEXT I
180  FOR I=1 TO M
190  FOR J=1 TO N
200  IF A[I,J]+B[I,J]=B[I,J]+A[I,J] THEN 230
210  PRINT "A+B <> B+A FOR MATRICES A AND B."
220  GOTO 260
230  NEXT J
240  NEXT I
250  PRINT "A+B=B+A FOR MATRICES A AND B."
260  END
RUN


WHAT IS THE ORDER OF THE MATRICES INVOLVED?4,3
A+B=B+A FOR MATRICES A AND B.

DONE


40 DATA 3,-7,1,.8,2,7,0,5
50 DATA 17,7,0,.2,3,3,10,5
RUN


WHAT IS THE ORDER OF THE MATRICES INVOLVED?4,2
A+B=B+A FOR MATRICES A AND B.

DONE
```

Micro Flow Chart

Exercise 6(b)




2
j←1
i←1
k←1
c_{ik}←0
c_{ik} IS THE ELEMENT IN ROW i, COLUMN k OF THE PRODUCT MATRIX A × B
j←1
c_{ik}←c_{ik} + (a_{ij} • b_{jk})
j←j+1
NO
j = n
YES
k←k+1
NO
k = m
YES
i←i+1
k←1
NO
i = m
YES
i←1
k←1
d_{ik}←0
d_{ik} IS THE ELEMENT IN ROW i, COLUMN k OF THE PRODUCT MATRIX B × A
j←1
d_{ik} ← d_{ik} + (b_{ij} • a_{jk})
j←j+1
NO
j = n
YES
k←k+1
NO
k = m
YES
i←i+1
k←1
NO
i = m
YES
3


3
i←1
j←1
OUTPUT
cij
NO
j←j+1
j = n
YES
NO
i←i+1
j←1
i = m
YES
i←1
j←1
OUTPUT
dij
NO
j←j+1
j = n
YES
NO
i←i+1
j←1
i = m
YES
STOP

Example Program

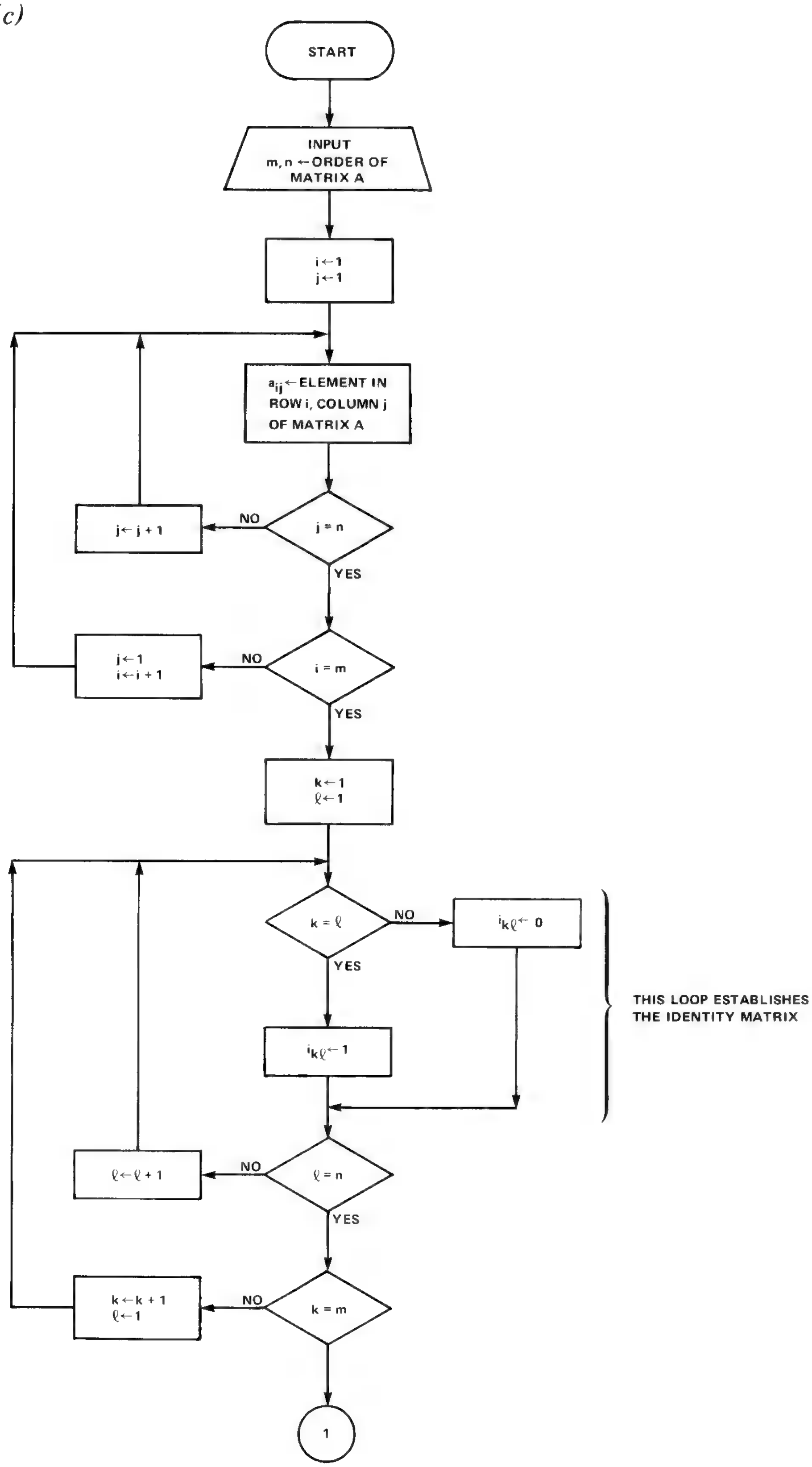
Exercise 6(b)

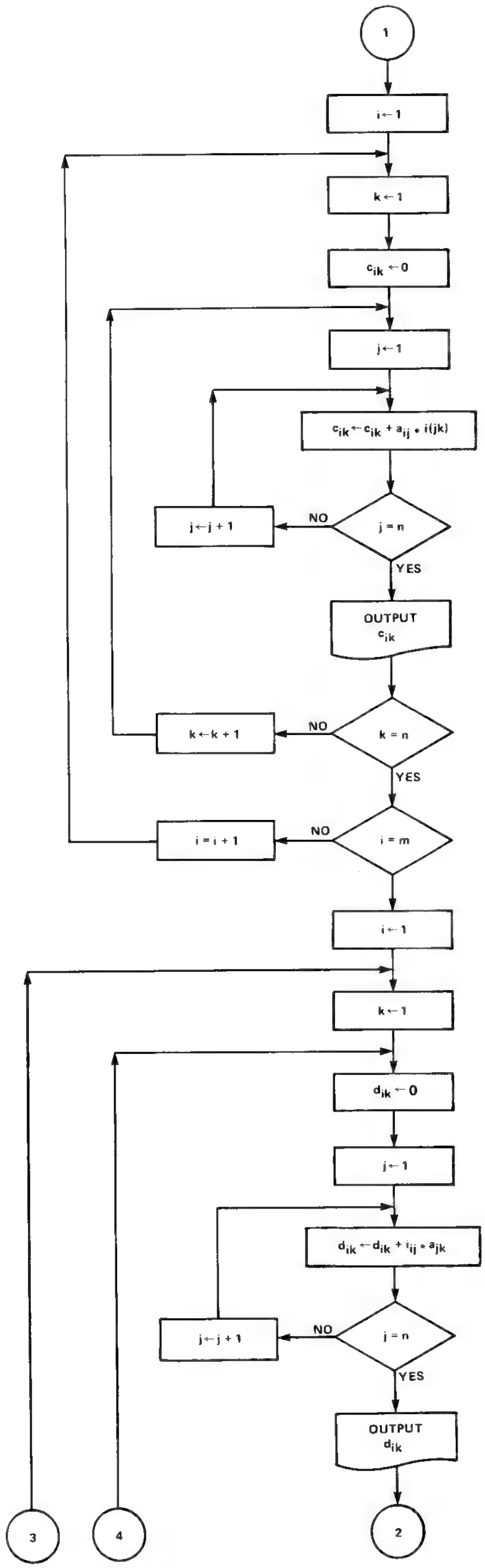
```
10  REM--THIS PROGRAM PRINTS OUT THE PRODUCTS A X B AND B X A
11  REM--FOR TWO MATRICES A AND B TO ILLUSTRATE THAT MATRIX
12  REM-- MULTIPLICATION IS NON COMMUTITATIVE. LINE 20 REQUIRES THE INPUT
13  REM--OF THE ORDER OF THE MATRICES.  THE DATA LINE SHOULD
14  REM--CONTAIN THE ELEMENTS OF A AND B IN THAT ORDER.
20  INPUT M,N
30  DATA 6,-2,1,0,1,-1,3,2,1,-2,3,0,4,5,3,-1,0,1
40  FOR I=1 TO M
50  FOR J=1 TO N
60  READ A[I,J]
70  NEXT J
80  NEXT I
90  FOR I=1 TO M
100  FOR J=1 TO N
110  READ B[I,J]
120  NEXT J
130  NEXT I
140  FOR I=1 TO M
145  FOR K=1 TO M
146  C[I,K]=0
150  FOR J=1 TO N
160  C[I,K]=C[I,K]+A[I,J]*B[J,K]
170  NEXT J
175  NEXT K
180  NEXT I
190  FOR I=1 TO M
195  FOR K=1 TO M
196  D[I,K]=0
200  FOR J=1 TO N
210  D[I,K]=D[I,K]+B[I,J]*A[J,K]
220  NEXT J
225  NEXT K
230  NEXT I
235  PRINT "A X B"
240  FOR I=1 TO M
250  FOR J=1 TO M
260  PRINT C[I,J];
270  NEXT J
280  PRINT
290  NEXT I
300  PRINT
310  PRINT
320  PRINT
325  PRINT "B X A"
330  FOR I=1 TO M
340  FOR J=1 TO N
350  PRINT D[I,J];
360  NEXT J
370  PRINT
380  NEXT I
390  END
```

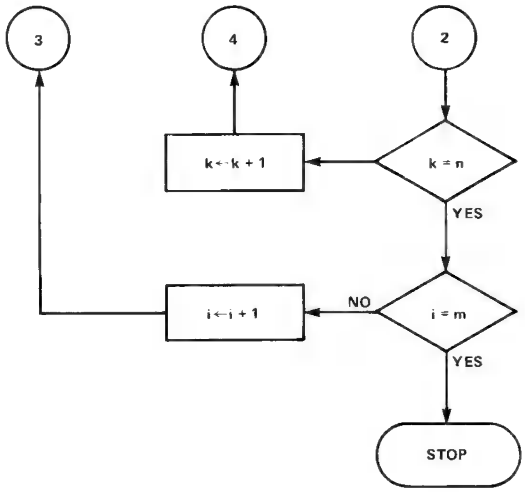
```
RUN

?2,2
A X B
 1        2
 3        4


B X A
 1        2
 3        4


30 DATA 6,-2,1,0,1,-1,3,2,1,-2,3,0,4,5,3,-1,0,1


?3,3
A X B
-21       8       -5
 5        5        2
 1        19       7


B X A
-12       7       -5
 33       3        2
-3        4        0

DONE
```

Micro Flow Chart

Exercise 6(c)




1
i←1
k←1
c_{ik} ←0
j←1
c_{ik}←c_{ik} + a_{ij} * i(jk)
j←j+1
NO
j = n
YES
OUTPUT
c_{ik}
k←k+1
NO
k = n
YES
i = i + 1
NO
i = m
i←1
k←1
d_{ik} ←0
j←1
d_{ik}←d_{ik} + i_{ij} * a_{jk}
j←j+1
NO
j = n
YES
OUTPUT
d_{ik}
3
4
2



Example Program

Exercise 6(c)

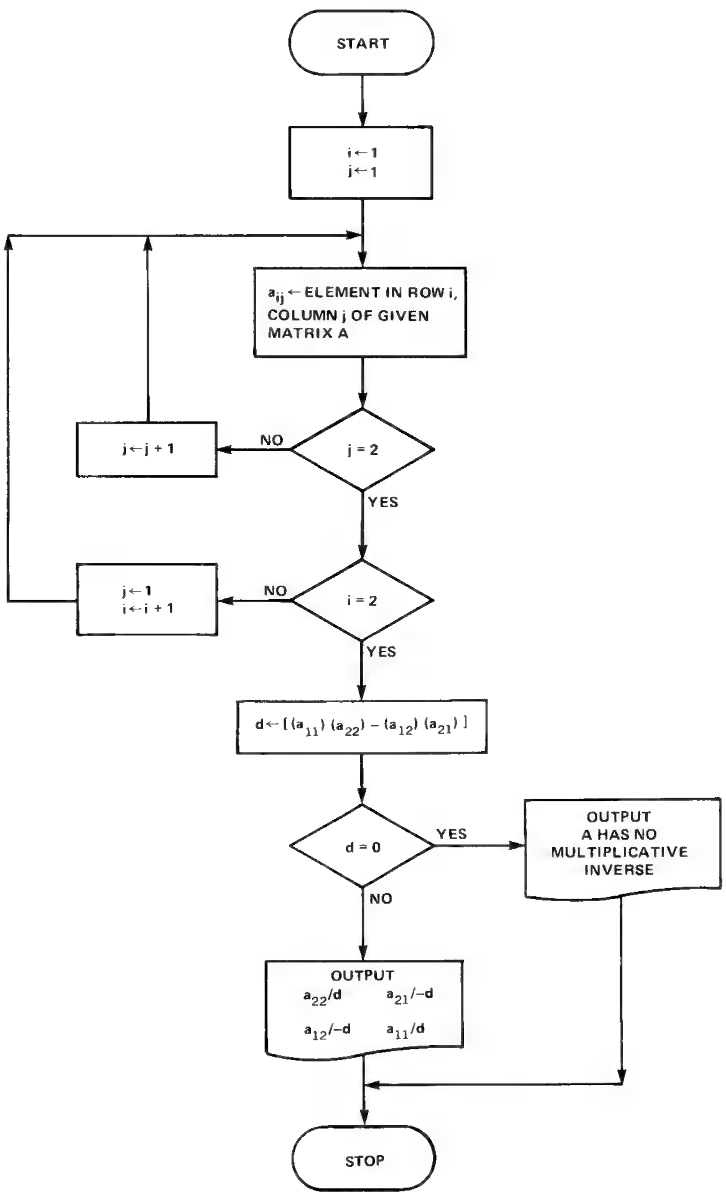
```
10  REM--A PROGRAM TO SHOW THAT FOR A GIVEN MATRIX OF REAL NUMBERS
11  REM--THERE EXISTS A MULPLICATIVE IDENTITY MATRIX.  ON THE
12  REM--DATA LINE ENTER THE ORDER AND ELEMENTS OF THE MATRIX BEING
13  REM--CONSIDERED.
20  DATA 4,4,1,-1,2,-2,3,-3,4,-4,5,-5,6,-6,7,-7,8,-8
30  READ M,N
40  FOR I=1 TO M
50  FOR J=1 TO N
60  READ A[I,J]
70  NEXT J
80  NEXT I
90  FOR K=1 TO M
100  FOR L=1 TO N
110  IF K=L THEN 125
120  LET I[K,L]=0
122  GOTO 130
125  LET I[K,L]=1
130  NEXT L
140  NEXT K
143  PRINT "A X I"
160  FOR I=1 TO M
170  FOR K=1 TO N
180  LET C[I,K]=0
190  FOR J=1 TO N
200  C[I,K]=C[I,K]+A[I,J]*I[J,K]
210  NEXT J
220  PRINT C[I,K];
230  NEXT K
240  PRINT
250  NEXT I
255  PRINT "I X A"
260  FOR I=1 TO M
270  FOR K=1 TO N
280  D[I,K]=0
290  FOR J=1 TO N
300  D[I,K]=D[I,K]+I[I,J]*A[J,K]
310  NEXT J
320  PRINT D[I,K];
330  NEXT K
340  PRINT
350  NEXT I
360  END
```

```
RUN

A X I
-17      0       22
 5      -.83            1
 6      -1       0


I X A
-17      0       22
 5      -.83            1
 6      -1       0

DONE

20 DATA 4,4,1,-1,2,-2,3,-3,4,-4,5,-5,6,-6,7,-7,8,-8
RUN

A X I
 1      -1       2      -2
 3      -3       4      -4
 5      -5       6      -6
 7      -7       8      -8


I X A
 1      -1       2      -2
 3      -3       4      -4
 5      -5       6      -6
 7      -7       8      -8

DONE
```

Micro Flow Chart

Exercise 6(d)

In computing the elements of the inverse matrix we get involved in round-off error. Consequently $A \times A^{-1}$ might not equal the identity matrix I. This problem opens the door for a discussion concerning computer error. The following program calls for the inverse matrix to be printed with the elements in decimal form and in rational form.



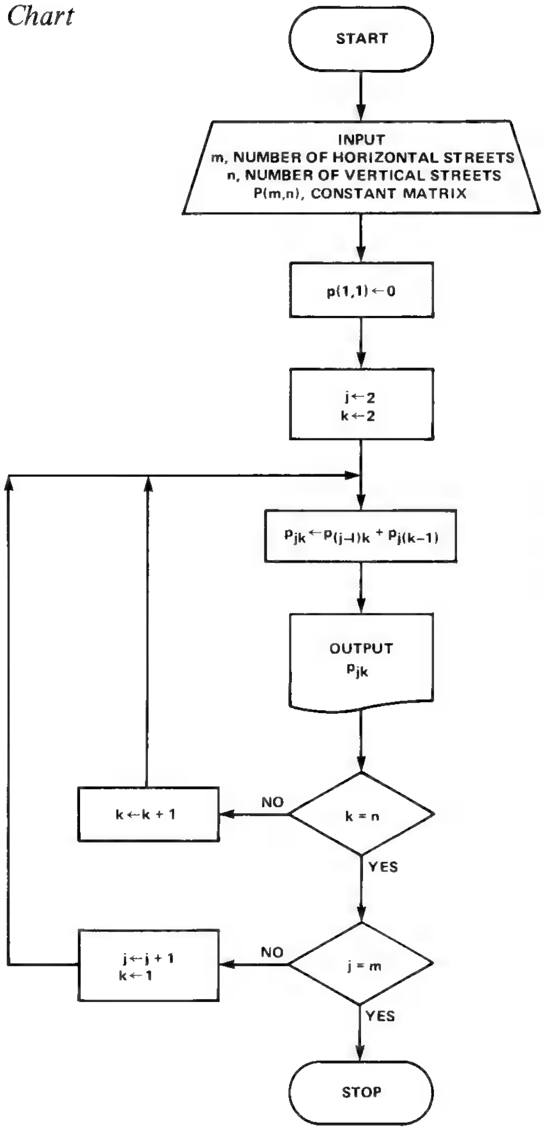
Example Program

Exercise 6(d)

```
10   REM-- A PROGRAM TO PRINT THE INVERSE OF A MATRIX
20   FOR I=1 TO 2
30   FOR J=1 TO 2
40   READ A[I,J]
50   NEXT J
60   NEXT I
70   D=A[1,1]*A[2,2]-A[1,2]*A[2,1]
80   IF D=0 THEN 160
90   PRINT A[2,2]/D;A[2,1]/-D
100   PRINT
110   PRINT A[1,2]/-D;A[1,1]/D
120   PRINT ""A[2,2];"/";D,A[2,1];"/";-D""
130   PRINT A[1,2]"/"-D,A[1,1]"/"D
140   PRINT
150   GOTO 20
160   PRINT "MATRIX 'A' DOES NOT HAVE AN INVERSE."
170   PRINT
180   GOTO 20
190   DATA 17,-3,2,1,6,-.5,36,-3
200   END


RUN

 4.34783E-02    -8.69565E-02

 .130435      .73913



 1     / 23        2     /-23

-3     /-23        17    / 23

MATRIX 'A' DOES NOT HAVE AN INVERSE.


OUT OF DATA  IN LINE 40
```

EXERCISE 7 – A Matrix Application

Micro Flow Chart



p_{jk} IS THE NUMBER OF PATHS BY WHICH YOU CAN ARRIVE AT THE INTERSECTION OF THE j^{th} HORIZONTAL STREET AND THE k^{th} VERTICAL STREET.

Example Program

Exercise 7

```
10  REM--A PROGRAM TO DETERMINE THE NUMBER OF PATHS FROM A POINT A
20  REM--TO A POINT B IN A CITY.  THE INPUT RECUIRES THE
30  REM--NUMBER OF HORIZONTAL STREETS AND THE NUMBER OF
40  REM--VERTICAL STREETS IN THAT ORER.
50  INPUT M,N
60  PRINT " THE NUMBER OF PATHS FROM P(1,1) TO:"
70  FOR I=1 TO M
72  FOR J=1 TO N
74  LET P[I,J]=1
76  NEXT J
78  NEXT I
80  LET P[1,1]=0
90  FOR J=2 TO M
100  FOR K=2 TO N
110  LET P[J,K]=P[J-1,K]+P[J,K-1]
120  PRINT TAB(5);"P(";J;",";K;") IS";P[J,K]
130  NEXT K
140  NEXT J
150  END
RUN

?4,4
 THE NUMBER OF PATHS FROM P(1,1) TO:
     P( 2     , 2     ) IS 2
     P( 2     , 3     ) IS 3
     P( 2     , 4     ) IS 4
     P( 3     , 2     ) IS 3
     P( 3     , 3     ) IS 6
     P( 3     , 4     ) IS 10
     P( 4     , 2     ) IS 4
     P( 4     , 3     ) IS 10
     P( 4     , 4     ) IS 20

DONE
```

```
RUN


?6,8
 THE NUMBER OF PATHS FROM P(1,1) TO:
     P( 2    , 2    ) IS 2
     P( 2    , 3    ) IS 3
     P( 2    , 4    ) IS 4
     P( 2    , 5    ) IS 5
     P( 2    , 6    ) IS 6
     P( 2    , 7    ) IS 7
     P( 2    , 8    ) IS 8
     P( 3    , 2    ) IS 3
     P( 3    , 3    ) IS 6
     P( 3    , 4    ) IS 10
     P( 3    , 5    ) IS 15
     P( 3    , 6    ) IS 21
     P( 3    , 7    ) IS 28
     P( 3    , 8    ) IS 36
     P( 4    , 2    ) IS 4
     P( 4    , 3    ) IS 10
     P( 4    , 4    ) IS 20
     P( 4    , 5    ) IS 35
     P( 4    , 6    ) IS 56
     P( 4    , 7    ) IS 84
     P( 4    , 8    ) IS 120
     P( 5    , 2    ) IS 5
     P( 5    , 3    ) IS 15
     P( 5    , 4    ) IS 35
     P( 5    , 5    ) IS 70
     P( 5    , 6    ) IS 126
     P( 5    , 7    ) IS 210
     P( 5    , 8    ) IS 330
     P( 6    , 2    ) IS 6
     P( 6    , 3    ) IS 21
     P( 6    , 4    ) IS 56
     P( 6    , 5    ) IS 126
     P( 6    , 6    ) IS 252
     P( 6    , 7    ) IS 462
     P( 6    , 8    ) IS 792

DONE
```

MODULAR ARITHMETIC

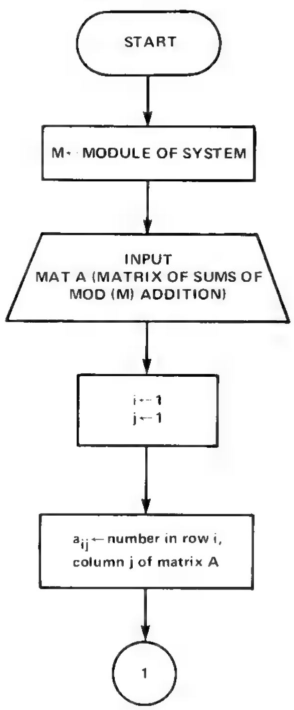
Modular arithmetic should be a satisfying topic of study because it is a finite system, which allows the student to come to concrete, provable conclusions concerning its properties. In the previous section we indicated that matrix theory has many applications. Therefore, it is perfectly acceptable to use the BASIC matrix instructions in the following exercises. Because some users of BASIC may not have these commands available to them the programs are written without them. However, on the listings where to replace statements with MAT statements are indicated.

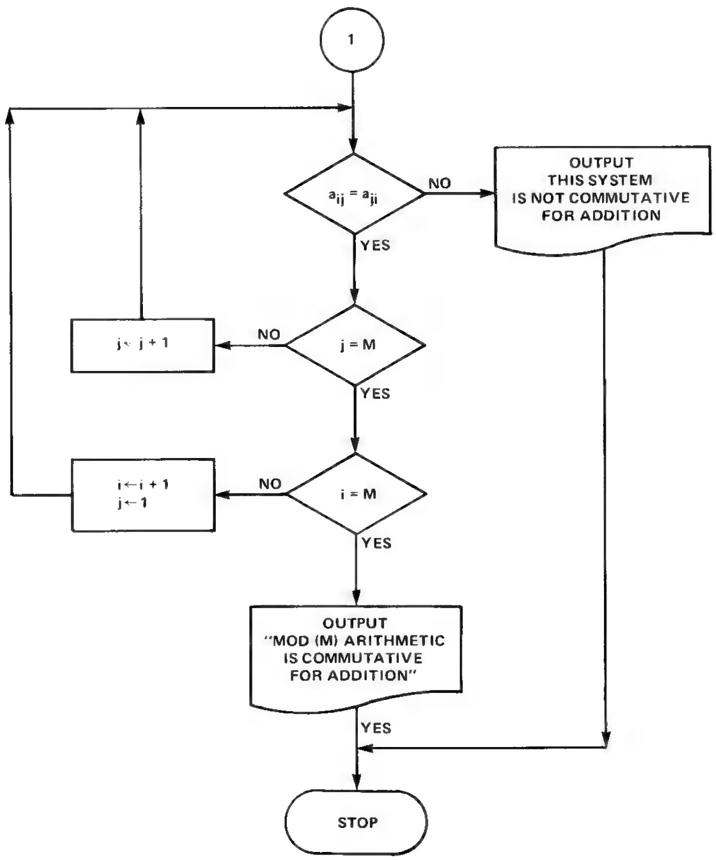
EXERCISE 8 – Commutativity of Mod (M) Arithmetic

The problem analysis and the flow chart in the student text are rather detailed and will probably need no further explanation

Micro Flow Chart

Exercise 8(a)




1
$a_{ij} = a_{ji}$
NO
OUTPUT
THIS SYSTEM
IS NOT COMMUTATIVE
FOR ADDITION
YES
$j \leftarrow j + 1$
NO
$j = M$
YES
$i \leftarrow i + 1$
$j \leftarrow 1$
NO
$i = M$
YES
OUTPUT
"MOD (M) ARITHMETIC
IS COMMUTATIVE
FOR ADDITION"
YES
STOP

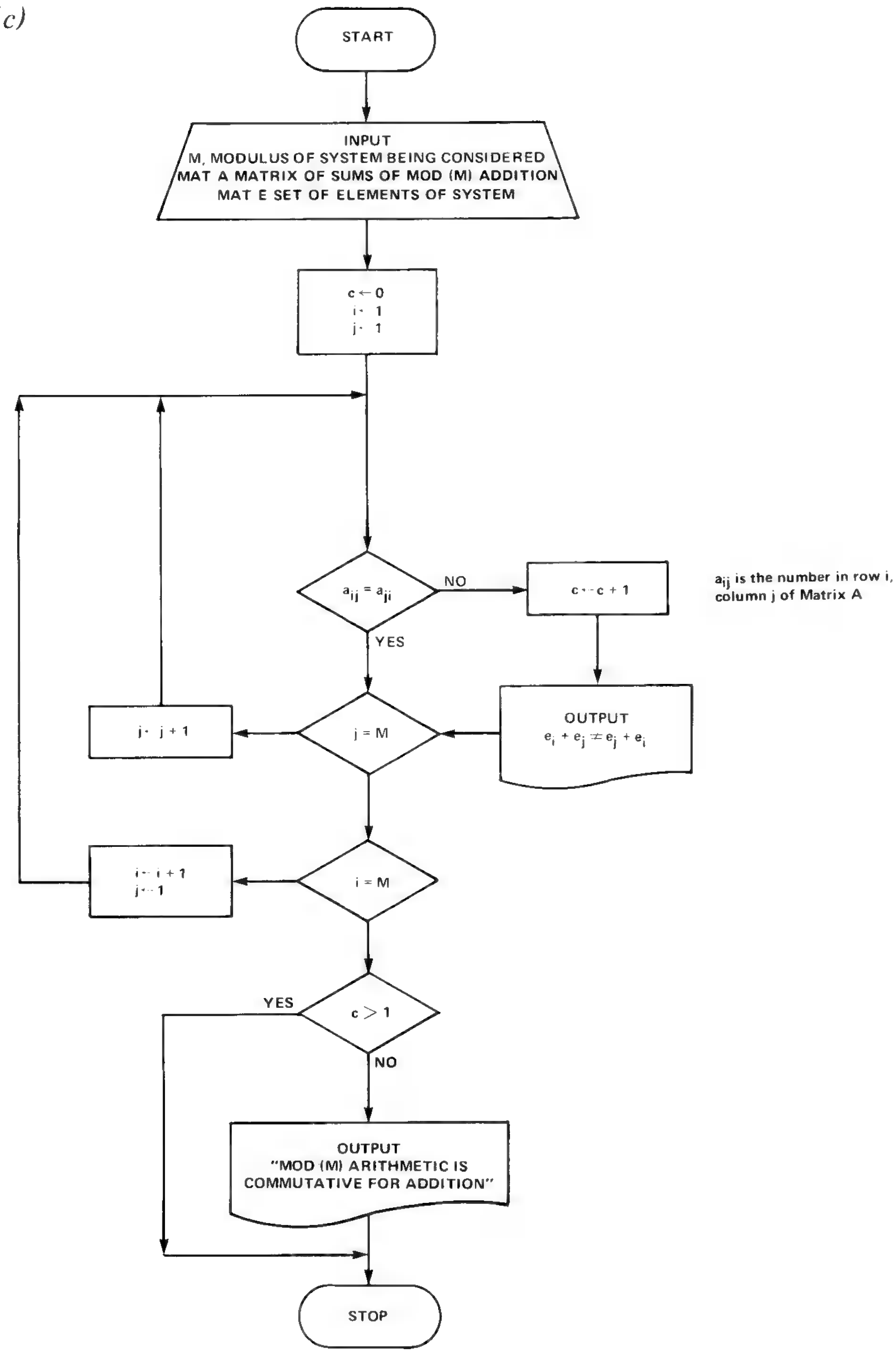
Example Program

Exercise 8(a) and 8(b)

```
1   REM--THIS PROGRAM WILL TEST COMMUNTATIVITY FOR ADDITION IN MODULAR
2   REM--ARITHMETIC.  THE DATA LINE SHOULD CONTAIN THE ELEMENTS FROM
4   REM--ROW 1 DOWN THROUGH THE LAST ROW. THE MATRIX WILL BE
5   REM--OF THE ORDER M, THEREFORE LINE 20 REQUIRES THE MODULE
6   REM--NUMBER BE INPUT FOR M.
20   INPUT M
30   FOR I=1 TO M
32   FOR J=1 TO M
34   READ A[I,J]      30   MAT   READ A[M,M]
36   NEXT J
38   NEXT I
40   DATA 0,1,2,3,1,2,3,0,2,3,0,1,3,0,1,2
50   FOR I=1 TO M
60   FOR J=1 TO M
70   IF A[I,J]=A[J,I] THEN 100
80   PRINT "THIS SYSTEM IS NOT COMMUTATIVE FOR ADDITION."
90   GOTO 130
100   NEXT J
110   NEXT I
120   PRINT "MOD "M" ARITHMETIC IS COMMUTATIVE FOR ADDITION."
130   END
RUN


?4
MOD   4      ARITHMETIC IS COMMUTATIVE FOR ADDITION.

DONE
40 DATA 0,1,2,3,1,2,3,0,2,3,0,1,3,0,3,2
RUN


?4
THIS SYSTEM IS NOT COMMUTATIVE FOR ADDITION.

DONE
```

Micro Flow Chart

Exercise 8(c)



Example Program

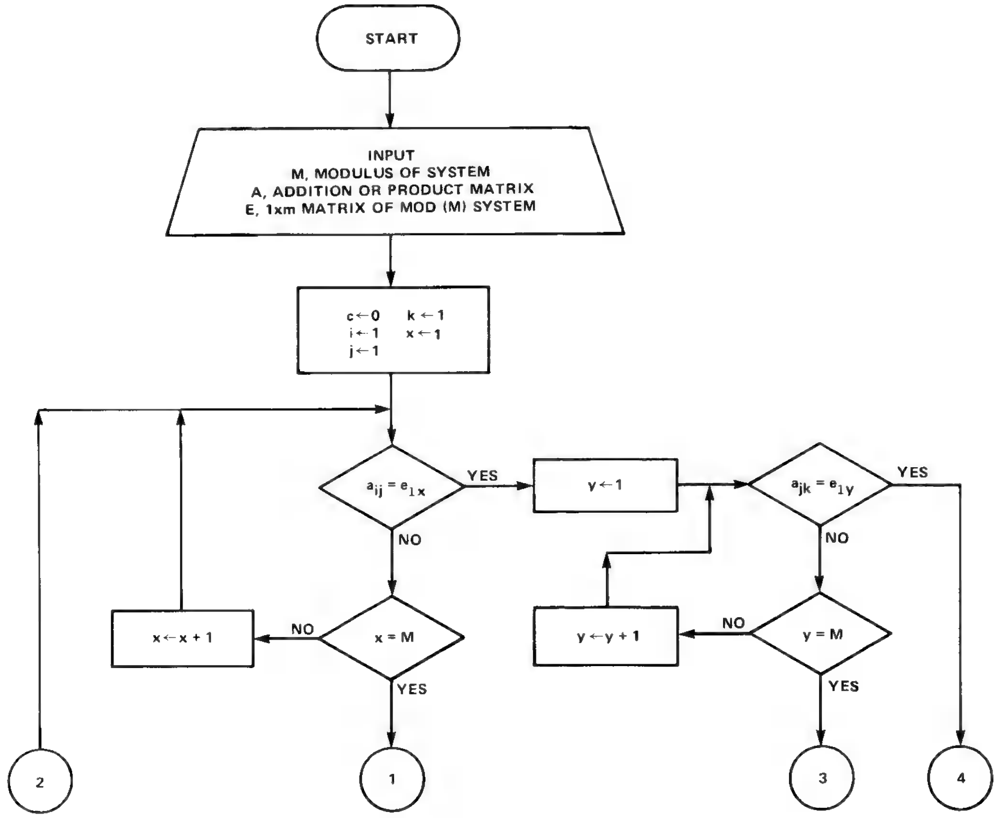
Exercise 8(c)

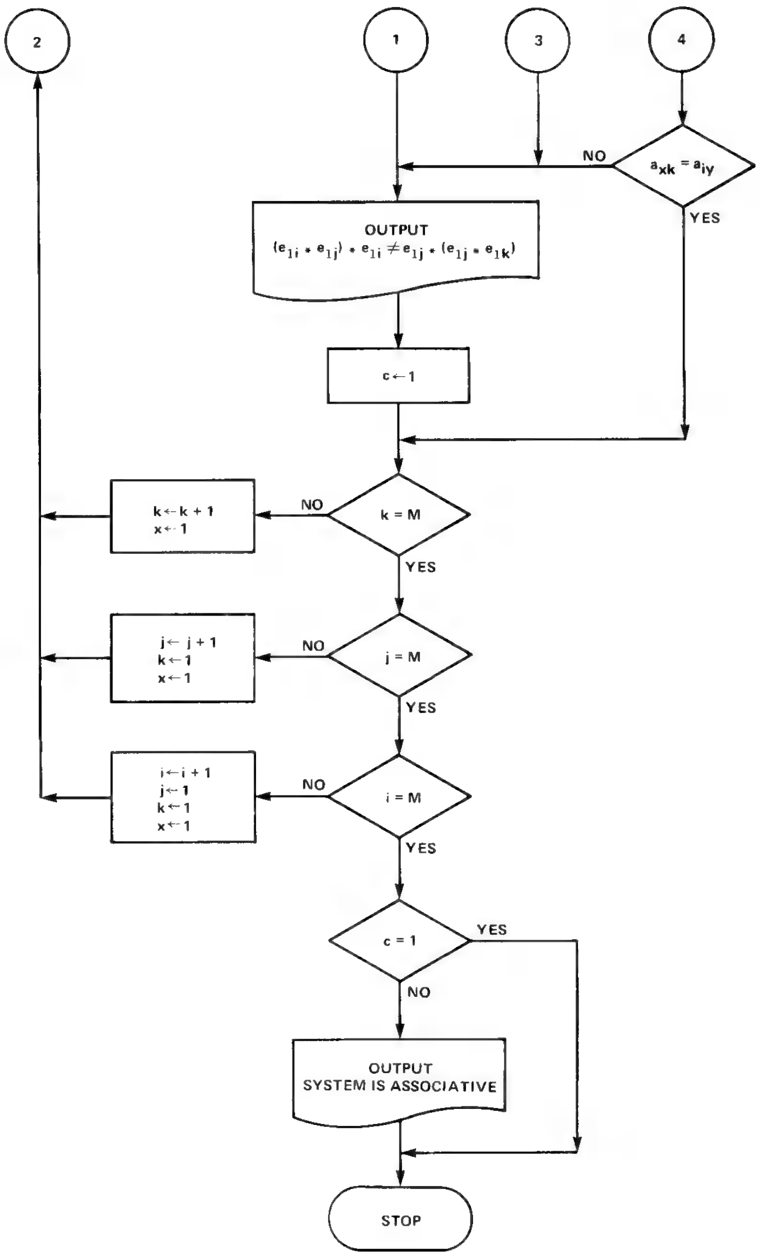
```
10   REM--THIS PROGRAM WILL TEST COMMUTATIVITY FOR ADDITION IN MODULAR
20   REM--ARITHMETIC.  ON THE DATA LINE ENTER THE ELEMENTS OF THE
30   REM--MOD IN ORDER FOLLOWED BY THE ELEMENTS OF THE ADDITION MATRIX
40   DATA 0,1,2,3,0,1,2,3,1,2,3,0,2,0,0,1,3,1,2,3
50   REM--BE OF THE ORDER M, THEREFORE LINE 20 REQUIRES THE
60   REM--MODULE NUMBER BE INPUT FOR M.
70   INPUT M
80   FOR I=1 TO M
81   READ E[1,I]       80   MAT   READ E[1,M]
82   NEXT I
83   FOR I=1 TO M
84   FOR J=1 TO M
85   READ A[I,J]       83   MAT   READ A[M,M]
86   NEXT J
87   NEXT I
90   DATA 0,1,2,3,0,1,2,3,1,2,3,0,2,3,0,1,3,0,1,2
100   C=0
110   FOR I=1 TO M
120   FOR J=1 TO M
130   IF A[I,J]=A[J,I] THEN 180
140   C=C+1
150   IF C>1 THEN 170
160   PRINT "THIS SYSTEM IS NOT COMMUTATIVE FOR THE FOLLOWING CASES:"
170   PRINT E[1,I];"+";E[1,J];"<>";E[1,J];"+";E[1,I]
180   NEXT J
190   NEXT I
200   IF C>1 THEN 230
210   GOTO 220
220   PRINT "MOD "M" ARITHMETIC IS COMMUTATIVE FOR ADDITION."
230   END
RUN


?4
MOD   4      ARITHMETIC IS COMMUTATIVE FOR ADDITION.

DONE
```

```
40 DATA 0,1,2,3,0,1,2,3,1,2,3,0,2,0,0,1,3,1,2,3
RUN


?4
THIS SYSTEM IS NOT COMMUTATIVE FOR THE FOLLOWING CASES:
 1      + 2      <> 2      + 1
 1      + 3      <> 3      + 1
 2      + 1      <> 1      + 2
 2      + 3      <> 3      + 2
 3      + 1      <> 1      + 3
 3      + 2      <> 2      + 3

DONE
```

EXERCISE 9 – Associativity for Mod (4) and (5) Arithmetic

Micro Flow Chart




2
1
3
4
NO
$a_{xk} = a_{iy}$
YES
OUTPUT
$(e_{1i} * e_{1j}) * e_{1i} \neq e_{1j} * (e_{1j} * e_{1k})$
$c \leftarrow 1$
$k \leftarrow k+1$
$x \leftarrow 1$
NO
k = M
YES
$j \leftarrow j+1$
$k \leftarrow 1$
$x \leftarrow 1$
NO
j = M
YES
$i \leftarrow i+1$
$j \leftarrow 1$
$k \leftarrow 1$
$x \leftarrow 1$
NO
i = M
YES
c = 1
YES
NO
OUTPUT
SYSTEM IS ASSOCIATIVE
STOP

Example Program

Exercise 9

```
10   REM--THIS PROGRAM MODELS THE ASSOCIATIVE PROPERTY FOR MODULAR
11   REM--ARITHMETIC (MULTIPLICATION OR ADDITION).  ENTER IN THE DATA
12   REM--LINE;  THE MODULE OF THE SYSTEM BEING CONSIDERED; THE
13   REM--ELEMENTS OF THE SYSTEM; AND THE ELEMENTS OF THE PRODUCT
14   REM--MATRIX.
20   DATA 4,0,1,2,3,0,0,0,0,0,1,2,3,0,2,0,2,0,3,2,1
30   READ M
70   FOR I=1 TO M
72   READ E[1,I]      70   MAT  READ E[1,M]
74   NEXT I
80   FOR I=1 TO M
81   FOR J=1 TO M
82   READ A[I,J]      80   MAT  READ A[M,M]
83   NEXT J
84   NEXT I
85   C=0
120   FOR I=1 TO M
130   FOR J=1 TO M
140   FOR K=1 TO M
150   FOR X=1 TO M
160   IF A[I,J]=E[1,X] THEN 260
170   NEXT X
180   PRINT "("E[1,I]"*"E[1,J]")*"E[1,K]"<>"E[1,I]"*("E[1,J];
181   PRINT "*"E[1,K]")"
190   C=1
200   NEXT K
210   NEXT J
220   NEXT I
230   IF C=1 THEN 320
240   PRINT "MOD(";M;")  ARITHMETIC IS ASSOCIATIVE FOR MULTIPLICATION"
250   GOTO 320
260   FOR Y=1 TO M
270   IF A[J,K]=E[1,Y] THEN 300
280   NEXT Y
290   GOTO 180
300   IF A[X,K]=A[I,Y] THEN 200
310   GOTO 180
320   END
```

```
RUN

MOD( 4      )  ARITHMETIC IS ASSOCIATIVE FOR MULTIPLICATION

DONE

20 DATA 5,0,1,2,3,4,0,0,0,0,0,0,1,2,3,4,0,2,4,3,3,0,3,1,4,2,0
21 DATA 4,3,2,1
RUN

( 2     * 2     )* 3     <> 2     *( 2     * 3     )
( 2     * 2     )* 4     <> 2     *( 2     * 4     )
( 2     * 3     )* 2     <> 2     *( 3     * 2     )
( 2     * 3     )* 3     <> 2     *( 3     * 3     )
( 2     * 3     )* 4     <> 2     *( 3     * 4     )
( 2     * 4     )* 2     <> 2     *( 4     * 2     )
( 3     * 2     )* 3     <> 3     *( 2     * 3     )
( 3     * 4     )* 3     <> 3     *( 4     * 3     )
( 4     * 2     )* 3     <> 4     *( 2     * 3     )
( 4     * 3     )* 3     <> 4     *( 3     * 3     )

DONE

20 DATA 5,0,1,2,3,4,0,0,0,0,0,0,1,2,3,4,0,2,4,1,3,0,3,1,4,2,0
21 DATA 4,3,2,1
RUN

MOD( 5      )  ARITHMETIC IS ASSOCIATIVE FOR MULTIPLICATION

DONE
```

EXERCISE 10 – Multiplicative Inverse for Mod (M) Arithmetic

Requiring students to model the Multiplicative Inverse Property (MIP) causes them to be aware of two conditions in the definition which are often overlooked or forgotten: (1) The additive identity element (zero) is not included in the MIP generalization, and (2) the generalization requires that each element have a unique *inverse.*

If the computer system your students are using has matrix manipulation capabilities, this exercise can make use of them.

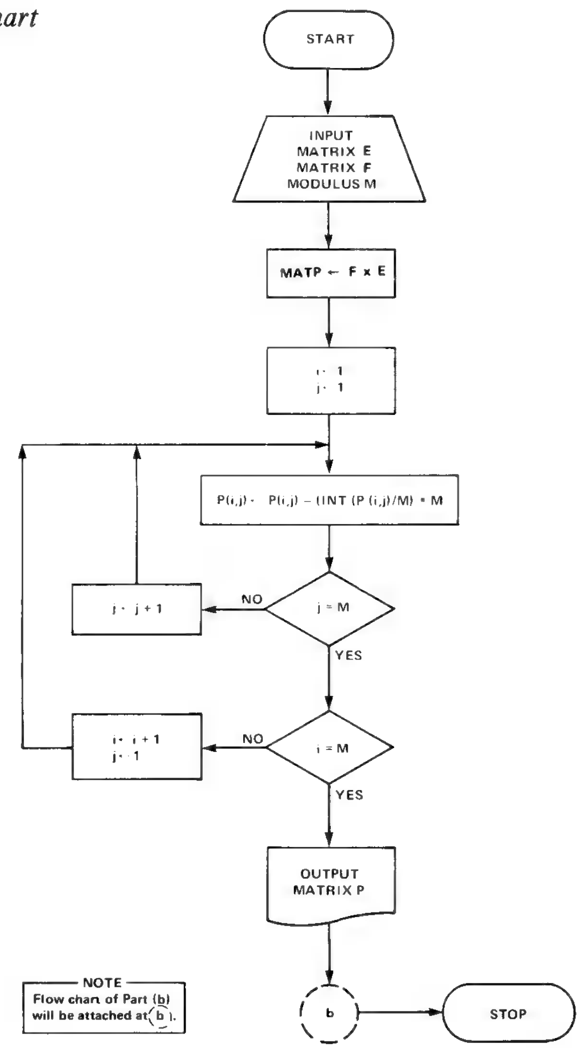
The following flow chart is written for a system with the matrix capabilities. There is a sample program for both systems.

To produce the product matrix P we will let Matrix $F = (0, 1, 2, \ldots M{-}1)$

and Matrix $E = \begin{pmatrix} 0 \\ 1 \\ 2 \\ \vdots \\ M{-}1 \end{pmatrix}$ *and* $P = E \times F$

Micro Flow Chart

Exercise 10(a)



Example Program

Exercise 10(a)

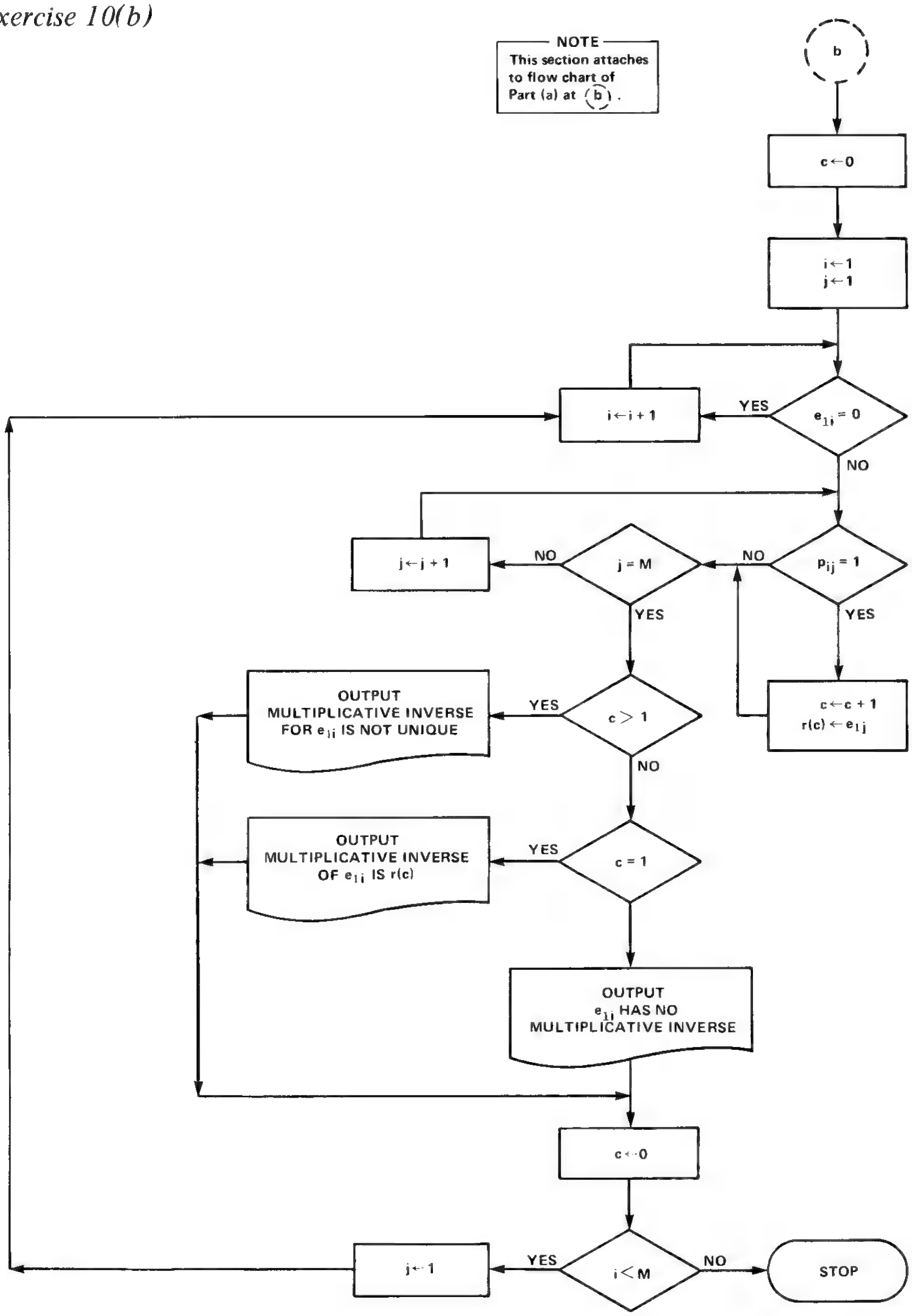
```
10  REM--THIS PROGRAM PRINTS THE PRODUCT MATRIX OF MOD (M)
20  REM--ARITHMETIC.  THE DATA LINE REQUIRES THE ELEMENTS FOR
30  REM--MATRIX F WHICH IS A 1 X M MATRIX AND MATRIX E, A M X 1
40  REM--MATRIX OF THE ELEMENTS OF MOD (M) SYSTEM.
50  DATA 4,0,1,2,3,0,1,2,3
60  READ M
70  FOR I=1 TO M
71  FOR J=1 TO M
72  LET P[I,J]=0     70  MAT P=ZER[M,M]
73  NEXT J
74  NEXT I
80  FOR I=1 TO M
82  READ E[1,I]      80  MAT  READ E[1,M]
84  NEXT I
90  FOR I=1 TO M
92  READ F[I,1]      90  MAT  READ F[M,1]
94  NEXT I
100  FOR I=1 TO M
102  FOR J=1 TO M
104  LET P[I,J]=P[I,J]+F[I,1]*E[1,J]   100  MAT P=F*E
106  NEXT J
108  NEXT I
110  FOR I=1 TO M
120  FOR J=1 TO M
130  P[I,J]=P[I,J]-(INT(P[I,J]/M)*M)
140  NEXT J
150  NEXT I
160  FOR I=1 TO M
162  FOR J=1 TO M
164  PRINT P[I,J],
166  NEXT J          160  MAT  PRINT P
167  PRINT
168  PRINT
169  NEXT I
170  END
```

```
RUN

 0              0              0              0

 0              1              2              3

 0              2              0              2

 0              3              2              1

DONE
```

```
50 DATA 5,0,1,2,3,4,0,1,2,3,4
RUN

 0              0              0              0              0

 0              1              2              3              4

 0              2              4              1              3

 0              3              1              4              2

 0              4              3              2              1

DONE
```

Micro Flow Chart

Exercise 10(b)



Example Program

Exercise 10(b)

```
10   REM--THIS PROGRAM WILL DETERMINE IF EACH ELEMENT OF A MOD (M)
20   REM--ARITHMETIC HAS A MULTIPLICATIVE INVERSE
30   DATA 3,0,1,2,0,1,2
40   READ M
50   FOR I=1 TO M
52   FOR J=1 TO M
54   LET P[I,J]=0       50  MAT P=ZER[M,M]
56   NEXT J
58   NEXT I
60   FOR I=1 TO M
62   READ E[1,I]        60  MAT  READ E[1,M]
64   NEXT I
70   FOR I=1 TO M
72   READ F[I,1]        70  MAT  READ F[M,1]
74   NEXT I
80   FOR I=1 TO M
82   FOR J=1 TO M
84   LET P[I,J]=P[I,J]+F[I,1]*E[1,J]    80  MAT P=F*E
86   NEXT J
88   NEXT I
90   FOR I=1 TO M
100   FOR J=1 TO M
110   P[I,J]=P[I,J]-(INT(P[I,J]/M)*M)
120   NEXT J
130   NEXT I
140   LET C=0
150   FOR I=1 TO M
160   FOR J=1 TO M
170   IF E[1,I]=0 THEN 330
180   IF P[I,J]=1 THEN 210
190   GOTO 230
200   IF J=M THEN 240
210   LET C=C+1
220   LET R[C]=E[1,J]
230   NEXT J
240   IF C>1 THEN 280
250   IF C=1 THEN 300
260   PRINT E[1,I];"HAS NO MULTIPLICATIVE INVERSE."
270   GOTO 310
280   PRINT "MULTPLICATIVE INVERSE OF"E[1,I]"IS NOT UNIQUE."
290   GOTO 310
300   PRINT "MULTIPLICATIVE INVERSE OF "E[1,I]" IS "R[C]
310   C=0
320   LET J=1
330   NEXT I
340   END
```

```
RUN

MULTIPLICATIVE INVERSE OF  1      IS  1
MULTIPLICATIVE INVERSE OF  2      IS  2

DONE


30 DATA 4,0,1,2,3,0,1,2,3
RUN

MULTIPLICATIVE INVERSE OF  1      IS  1
 2    HAS NO MULTIPLICATIVE INVERSE.
MULTIPLICATIVE INVERSE OF  3      IS  3

DONE


30 DATA 5,0,1,2,3,4,0,1,2,3,4
RUN

MULTIPLICATIVE INVERSE OF  1      IS  1
MULTIPLICATIVE INVERSE OF  2      IS  3
MULTIPLICATIVE INVERSE OF  3      IS  2
MULTIPLICATIVE INVERSE OF  4      IS  4

DONE


30 DATA 6,0,1,2,3,4,5,0,1,2,3,4,5
RUN

MULTIPLICATIVE INVERSE OF  1      IS  1
 2    HAS NO MULTIPLICATIVE INVERSE.
 3    HAS NO MULTIPLICATIVE INVERSE.
 4    HAS NO MULTIPLICATIVE INVERSE.
MULTIPLICATIVE INVERSE OF  5      IS  5

DONE
```

NOTES

NOTES

NOTES